Federal University of Pernambuco
Technology and Geosciences Center
Civil Engineering Graduate Department

Artur Castiel Reis de Souza

A MsCV framework using a non-orthodox MPFA-D for the simulation of two-phase flows on truly unstructured grids

Recife

2018

Artur Castiel Reis de Souza

A MsCV framework using a non-orthodox MPFA-D for the simulation of two-phase flows on truly unstructured grids

Thesis submitted to the faculty of the Graduate Course in Civil Engineering of the Federal University of Pernambuco as part of the requirements necessary to obtain a Master's degree in Civil Engineering. Supervisor:

Supervisor: Paulo Roberto Maciel Lyra, PhD
Co-supervisor: Darlan Karlo Elisiário de Carvalho, PhD

Federal University of Pernambuco – UFPE

Civil Engineering Graduate Program

Simulation and Management of Oil Reservoirs

Recife

2018

Catalogação na fonte
Bibliotecária: Rosineide Mesquita Gonçalves Luz / CRB4-1361 (BCTG)

S729m Souza, Artur Castiel Reis de.
A MsCV framework using a non-orthodox MPFA-D for the simulation of twophase flows on truly unstructured grids/ Artur Castiel Reis de Souza. – Recife, 2018
105 p. : il. (algumas color.) ; 30 cm. gráf. e tab.

Orientador: Prof. Dr. Paulo Roberto Maciel Lyra.

Dissertação (Mestrado) – Universidade Federal de Pernambuco. CTG. Programa de Pós-Graduação em Engenharia Civil, 2018.
Inclui referências e apêndice.

1. Engenharia Civil. 2. Multiescala. 3. Volumes finitos. 3. MPFA. 4. MsFV. 5. Malhas não-estruturadas. I. Lyra, Paulo Roberto Maciel (Orientador). II. Título.

624 CDD (22. Ed.) UFPE-BIBCTG/2018-86

UNIVERSIDADE FEDERAL DE PERNAMBUCO
PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA CIVIL

A comissão examinadora da Defesa de Dissertação de Mestrado

A MsCV FRAMEWORK USING A NON-ORTHODOX MPFA-D FOR THE SIMULATION OF TWO-PHASE FLOWS ON TRULY UNSTRUCTURED GRIDS

defendida por

Artur Castiel dos Reis Souza

Considera o candidato APROVADO

Recife, 29 de janeiro de 2018

Orientador - Prof. Dr. Paulo Roberto Maciel Lyra – UFPE
Coorientador - Prof. Dr. Darlan Karlo Elisiário de Carvalho - UFPE

Banca Examinadora:

_____
Prof. Dr. Paulo Roberto Maciel Lyra – UFPE
(orientador)

_____
Prof. Dr. Fernando Raul Licapa Contreras – UFPE
(examinador externo)

_____
Prof. Dr. Alessandro Romário Echevarria Antunes– UFPE
(examinador externo)

_____
Prof. Dr. Ramiro Brito Willmersdorf – UFPE
(examinador interno)

Dedicado aqueles que mais me inspiram, meu pai e minha mãe.

Acknowledgements

The author of this thesis would like to thank all those who supported me through the countless days of hard work and sacrifices. I offer a special thanks to the following groups:

The Science Support Foundation of the State of Pernambuco (FACEPE) and the Computer Modelling Group (CMG), without their funding this research would not be possible.

To all my friends at the High Performance Processing in Computational Mechanics group (PADMEC) for the many hours spent on the philosophical discussions on the subject of this thesis.

To my advisor and to my co-advisor, Professor Paulo Lyra and Professor Darlan Karlo for the opportunity to work on such an amazing field, for the guidance and friendship.

To my brother and my sister who always cheered my up and encouraged in moments when I struggled.

To my parents who were always there for me, supporting my decision to follow my dreams.

To my beloved great-aunt Ana who is always ready to listen me.

To my maternal great parents that even far are always so present in my life.

To my paternal great parents who have passed away but have left us their inexhaustible love.

I really hope I made you all proud!

"We wish to pursue the truth no matter where it leads — but to find the truth, we need imagination and skepticism both. We will not be afraid to speculate, but we will be careful to distinguish speculation from fact. The cosmos is full beyond measure of elegant truths; of exquisite interrelationships; of the awesome machinery of nature. The surface of the Earth is the shore of the cosmic ocean. On this shore we've learned most of what we know. Recently we've waded a little way out, maybe ankle deep, and the water seems inviting. Some part of our being knows this is where we came from. We long to return. And we can. Because the cosmos is also within us. We're made of star-stuff. We are a way for the cosmos to know itself."

(Carl Sagan, Cosmos: A Personal Voyage)

Abstract

Modern geocellular models may contain up to hundreds million cells, while practical petroleum reservoir models handle at most a fraction of this quantity turning the direct numerical simulation of multiphase flow in heterogeneous and anisotropic medium infeasible. To overcome these problems Multiscale Finite Volume Methods (MsFVM) uses restriction algorithm to transfer information onto a lower-resolution grid, solve the resulting coarse system and by using a set of basis function, project back the solution onto the higher-resolution grid. Nonetheless, the MsFVM basis function fail to deal with high-resolution geological properties on general grids as they often rely on TPFA, which is only consistent for k-orthogonal grids. Furthermore, MsFVM possess no framework capable of generating the geometric entities needed for simulation on unstructured coarse-scale meshes. The Multiscale Restricted Smoothed Basis (MsRSB) method creates this framework and expands the multiscale approach to unstructured coarse meshes. However, it fails to produce consistent solutions on fine-scale unstructured meshes and for arbitrary permeability tensors as it also uses TPFA. In this thesis, we couple a Multi-Point Flux Approximation (MPFAD) with a Diamond stencil to the MsRSB to extend its use to general unstructured grids. The resulting framework showed prominent results producing accurate solutions for two-phase flow simulation in heterogeneous and mildly anisotropic medium with unstructured grids on a coarse and a fine scale.

Keywords: Multiscale. MsFV. Finite volume. MPFA. Unstructured-grids.

Resumo

Hoje em dia, os modelos geo-celulares podem conter até centenas de milhares de células, enquanto os modelos de reservatórios de petróleo trabalham no máximo com uma pequena fração desta quantidade. Isto torna a simulação numérica de escoamento multifásico em meios heterogêneos e anisotrópicos nestas malhas inviáveis. Para superar estas limitações, o Método dos Volumes Finitos Multiescala (MsFVM) usa algoritmos de restrição para transferir informação da escala de malha de alta-resolução para uma malha de menor resolução onde o sistema de equações é resolvido para então, utilizando um conjunto de funções de base, projetar a solução de volta na malha de maior resolução. No entanto, as funções de base do MsFVM são incapazes de capturar informações geológicas em alta-definição em malhas não estruturadas uma vez que o método depende de uma aproximação do fluxo do tipo por dois pontos (TPFA) que é consistente apenas com malhas k-ortogonais. Além disso, o MsFVM não possui um framework capaz de gerar as entidades necessárias para a simulação em malhas não estruturadas na escala de baixa resolução. O Métodos Multiescala das Funções de Base Restritamente Suavizadas (MsRSB) criar este metodologia e expande os conceitos do método clássico para permitir a simulação em malhas de baixa resolução não estruturadas. Contudo, este método falha em produzir soluções consistentes para malhas de alta resolução não estruturadas, uma vez que ele também depende de uma aproximação do tipo TPFA. Nesse trabalho, nós acoplamos o MsRSB com uma aproximação do fluxo do tipo multipontos com estêncil diamante para estender o seu uso para malhais não estruturadas em geral.

Palavras-chave: Multiescala. MsFV. Volumes finitos. MPFA. Malhas não-estruturadas.

List of Figures

List of Tables

List of abbreviations and acronyms

TPFA	Two-Point Flux Approximation
MPFA	Multi-Point Flux Approximation
MPFA-D	Multi-Point Flux Approximation with a Diamond stencil
PDE	Partial Derivative Equation
CV	Control Volume
CVD	Control-Volume Distributed
FV	Finite Volume
BC	Boundary Conditions
MsFV	Multiscale Finite Volume
MsFVM	Multiscale Finite Volume Method
MsRSB	Multiscale Finite Volume
MsRSB	Multiscale Finite Volume Method
MsCV	Multiscale Control Volumed
MsMFEM	Multiscale Mixed Finite Element Method
MsFEM	Multiscale Finite Element Method
GP	Geometric Partitioner
PVI	Pore Volumes Injected
IMPES	Implicit Pressure Explicit Saturation

Contents

1 INTRODUCTION

The advances in geostatistical modeling and characterization allows information from different scales to be integrated in order to generate geocellular models whose resolution typically range from 10^8 to 10^9 blocks, meanwhile the standard models of flow simulation in porous media can handle 10^6 to 10^7 blocks (ZHOU, 2010). In this way, multiple direct simulations on these high-resolution grids become infeasible. To overcome this limitation, scale-transferring methods have been devised. In essence, they allow high-resolution geostatistical data to be integrated onto the flow simulation grid (BARBOSA, 2017).

Among them, two branches of schemes stand out: the upscaling and the multiscale methods (DURLOFSKY, 2005). The first generally employs a sort of homogenization (Farmer, 2002), even when there is no formal separation between the scales. In these schemes a solution is found at the coarse-scale space leading to fast results, which often are unable to preserve details of the physical properties causing deterioration of the accuracy of the representation of the studied phenomena.

On the other hand, the multiscale schemes develop a set of operators, which are capable of projecting the fine-scale discrete system onto the coarse-scale space, this low-resolution system is then solved and the multiscale operator projects the solution back to the high-resolution grid (Hou and Wu,1997; Jenny, Lee and Tchelepi, 2006; Zhou and Tchelepi,2008). This preserves the natural coupling between scales avoiding inconsistencies and loss of fine-scale information inherent to most upscaling methods.

The central idea and core of all multiscale is the basis functions, a set of functions devised to exchange information between high-resolution and low-resolution grids. The Multiscale Finite Element Method (MsFEM) propposed by Hou and Wu (1997) for elliptical problems solves these functions using a set of boundary conditions to uncouple each coarse volume. Therefore, the velocity field is not fully conservative as the basis functions neglect flux normal on the surface of the coarse volumes. Using a Multiscale Mixed Finite Element Methods (MsMFEM), Chen and Hou (2002) dealt with this problem by imposing mathematically flow conservation by creating basis functions that compute simultaneously the pressure and velocity. Arbogast and Bryant (2002) brought this scheme to the petroleum context as they adapted it for two-phase flow problems.

The Multiscale Finite Volume (MsFV) proposed by Jenny, Lee and Tchelepi (2003) uses a different strategy to reimpose flux conservation. In this method, the basis functions are calculated on auxiliary an auxiliary dual coarse mesh. Thus, the basis

function become non-conservative on the surface of each dual coarse volumes but on the fine-scale volumes inside them. The MsFV uses the flow on the surfaces of each primal coarse volume to compute a new set of basis functions used to reconstruct a conservative velocity field. Although the use of another mesh increases the amount of data stored, it eliminates considerably the number of degrees of freedom by removing the need to calculate an operator that simultaneously integrates information of pressure and velocity fields. Jenny, Lee and Tchelepi (2006) extended this method to work with incompressible flow on three-dimensional domain. Additionally, JENNY; LEE; TCHELEPI studied techniques to improve the performance of Implicit Pressure Explicit Saturation (IMPES) strategy used by MsFV. Zhou and Tchelepi (2008), Zhou (2010) developed an algebraic notation to write the MsFV and geral multiscale methods as a series of simple matrix operations. For that, ZHOU; TCHELEPI created the two scale-transferring operators: Restriction Operator and a Prolongation Operator.

Nevertheless, most of the methods in MsFVM family are unable to properly work on unstructured grids. This is due to three different factors: the use of Two-Point Flux Approximation (TPFA), which is only consistent for k-orthogonal grids, the difficulties presented in generating geometrical entities, such as auxiliary grids and coarse cell centers; and in calculating multiscale operators accordingly. In this context, a great effort has been made in the development of schemes capable of enabling the simulation on unstructured grids on both scales. Krogstad et al. (2009) developed a variant of MsMFEM capable of working with coarse meshes whose volumes are nearly degenerated and unstructured.

Moyner and Lie (2013) extended the MsFV by creating a framework to allow the simulation on unstructured grids on the coarse-scale. The Multiscale Restriction-Smoothed Basis (MsRSB) method devised by MOYNER; LIE calculate the basis functions interactively smoothing the indicator function of each coarse volume restricted to corresponding support region giving birth to a robust and fast scheme. However, it is also limited to the simulation on unstructured coarse-scale meshes as it uses a TPFA.

On the other hand, many authors have studied methods to overcome the limitations of the classical TPFA. Among them, the work of Edwards and Rogers (1998) and Aavatsmark et al. (1998) stand out. Separately, they generalized the control volume distributed multi-point flux approximation (CVD-MPFA) devised by Crumpton, Shaw and Ware (1995) to work on general unstructured grids. These strategies enabled direct simulation on unstructured grids with full permeability tensors. Nonetheless they may fail to converge for high anisotropic media on highly distorted meshes (EDWARDS; ZHENG, 2008; CHEN et al., 2008).

In this thesis, we replace the TPFA used on the MsRSB for the approximation of the basis functions by a non-orthodox MPFA with a diamond stencil (MPFA-D) to

create the Multiscale Control-Volumed (MsCV) framework to allow the simulation on fine and coarse scales unstructured grids.

1.1 Research Objectives

In this context, the general objective of this thesis is the study and development of multiscale schemes for the 2-D numerical simulation of two-phase flows in heterogeneous and anisotropic petroleum reservoirs using truly untructured meshes.

1.2 Specific Objectives

1. To develop a multiscale framework for the simulation of one-phase and two-phase flows in highly heterogeneous and anisotropic petroleum reservoir using a non-orthodox flux approximation consistent with unstructured grids in any scale.
2. Incorporate a MPFA-D in a multiscale pressure solver
3. To create alternative tools to coarsen fine-scale grids.

1.3 Thesis Organization

This thesis is comprised of 6 different chapters. This first chapter is focused on introducing and reviewing the main issues of the simulation of two-phase flow in porous media using multiscale schemes. The second chapter consists of the mathematical formulation used to derive the PDEs that described two-phase flow in porous media. The third chapter refers to the numerical formulation, in which, we present the numerical schemes used to derive the discrete form of the equations of the previous chapter. As the multiscale strategy consists in the core of the thesis, the forth chapter is dedicated to explain the multiscale formulation along the creation of the entities that comprised the MsCV framework. The fifth chapter is a compilation of the results for single-phase and two-phase problem. Finally, the sixth and final chapter we present the conclusions and suggestions for further work.

2 MATHEMATICAL FORMULATION

This chapter gives a brief introduction on the laws and assumptions used to derive the equations that governs oil and water flow inside the reservoir rock. First, we present the equation that describe multiple phases flow in porous medium, Darcy's Law, following by mass conservation PDE that model pressure distribution inside the Oil Reservoir along with hyperbolic PDE that models the way fluid phases are transported.

2.1 Darcy's Law

Darcy's law was originally formulated by Henry Darcy based on the results of a experiment on water flow through bed sands. While the Navier-Stokes Equation describe flow in a microscopic level, Dary's law is a volume averaging approximation of the Navier-Stokes equation (ERWING, 1983) on scale that is small as possible but coarse enough to have an average that capture well the heterogeneity of the porous medium.

Darcy's law states the velocity of a phase is linearly proportional to the pressure gradient, this can be described with the equation bellow:

$$\vec{v_i} = -\lambda_i \underline{\underline{K}} \vec{\nabla} p_i \quad \text{with} \quad \lambda_i = k_{ri}/\mu_i, \quad i = \text{water (w), oil (o) in } \Omega \tag{2.1}$$

where λ_i , k_{ri} and μ_i are respectively mobility , relative permeability and viscosity of the phase i.

For a two-dimensional space ($\mathbb{R}^2$) using a Cartesian coordinate system, the absolute permeability tensor $\underline{\underline{K}}$ is defined as:

$$\underline{\underline{K}}(\vec{x}) = \begin{bmatrix} k_{xx} & k_{xy} \\ k_{yx} & k_{yy} \end{bmatrix} \quad \forall \vec{x} \in \mathbb{R}^2 \tag{2.2}$$

where $\vec{x} = (x, y)$.

The absolute permeability $\underline{\underline{K}}$ is an intrinsic property and exclusive to the rock reservoir. It measures the capacity of a medium to allow flow. On the other hand, the relative permeability k_{ri}, is a dimensionless measure of the effective permeability of that phase.

Therefore, it is necessary to have an equation that models the influence of one phase flow to the other phases. The Brooks and Corey model used in this work is a constitutive equation (HELMIG, 1997) based on physical experiments defined by the

following expression:

$$k_{rw} = \left(\frac{S_w - S_{wi}}{1 - S_{wi} - S_{or}} \right)^{n_w} \quad \text{and} \quad k_{ro} = \left(\frac{1 - S_w - S_{or}}{1 - S_{wi} - S_{or}} \right)^{n_o} \tag{2.3}$$

where S_i the fraction of the pore volume occupied by the phase $i = o, w$, S_{wi} and S_{or} is the residual saturation of the water phase and oil phases and n_w and n_o take different values (CHEN; HUAN; MA, 2006).

There are many other models for the relative permeability such as the van Genuchten, Yanosik and McCracken that can be found (CARVALHO, 2005; CHEN; HUAN; MA, 2006). This work uses exclusively the Brooks and Corey model for relative permeability.

2.2 Mass Conservation Equation

Let us consider an immiscible two-phase flow on a porous medium where the wetting and no-wetting phase are respectively water and oil. The medium is fully saturated i.e. all porous are full occupied by either one of the phases. The rock matrix and all fluids are consider incompressible. There is no mass transference between phases nor changes in temperature of the reservoir. We disregard capillary effect and we assume that no chemical reaction take place. Taking into account these assumptions we can apply mass conservation law to each phase and after some algebraic manipulation derive the equation bellow (CHEN; HUAN; MA, 2006):

$$\frac{\partial(\phi \rho_i S_i)}{\partial t} = -\nabla \cdot (\rho_i \vec{v_i}) + q_i \quad \text{in} \quad \Omega \times [0, t] \quad \text{where} \quad i = \text{w,o} \tag{2.4}$$

In (2.4), ϕ denotes the porosity of the medium, t the time and q_i represents source sink term that in this context can be a representation of injection and production wells, $\vec{v_i}$ phase velocity and ρ_i the density of the phase. We also assume that the medium is fully saturated, i.e.:

$$S_o + S_w = 1 \tag{2.5}$$

2.3 Water-Oil Immiscible Flow

From equation (2.4) under the assumptions stated in beginning of section 2.2 we can derive the two partial differential equations (PDE)s that govern water-oil flow in porous media. The first is the Pressure equation, an elliptic PDE, which provides the pressure distribution inside the oil reservoir. The latter is the Saturation equation, a

hyperbolic PDE, that shows the distribution of phase i inside the reservoir. Once the saturation of the phase i is found the other phase saturation can be found using the closure equation (2.5), also defined in the previous section.

2.3.1 Pressure Equation

The pressure equation is the elliptic PDE that governs water-oil flow in porous medium, and is obtained by combining and manipulating equations (2.1) and (2.4) for each phase. (CARVALHO, 2005; CHEN; HUAN; MA, 2006). It is expressed as:

$$-\vec{\nabla}\cdot\vec{v} = Q \quad \text{with} \quad \vec{v} = -\lambda\underset{\sim}{K}(\vec{\nabla}p - \rho_{\text{avg}}\vec{g}) \quad \text{in} \quad \Omega \tag{2.6}$$

where $\vec{v} = \vec{v}_w + \vec{v}_o$ is the total velocity of the fluid neglecting compressibility, ∇p is the pressure gradient and $Q = Q_w + Q_o$ is the sum of each phase volumetric source and sink terms divided by its density, $Q_i = q_i/\rho_i$. The average density ρ_{avg} is the a mobility weighted average. (HURTADO, 2011)

$$\rho_{\text{avg}} = \frac{\lambda_o\rho_o + \lambda_w\rho_w}{\lambda} \quad \text{with} \quad \lambda = \lambda_o + \lambda_w \tag{2.7}$$

We also neglect the gravity term on the Darcy's velocity equation (2.6) and find a simpler form for the total velocity:

$$\vec{v} = -\lambda\underset{\sim}{K}\vec{\nabla}p \tag{2.8}$$

By inserting this simplified velocity (2.8) inside (2.6), we obtain a global pressure equation for two-phase flow with no gravity:

$$\vec{\nabla}\cdot(\lambda\underset{\sim}{K}\vec{\nabla}p) = Q \tag{2.9}$$

Equation (2.9) can be also used to describe single-phase flow by simply setting $\lambda = 1$, and the total velocity becomes fluid velocity.

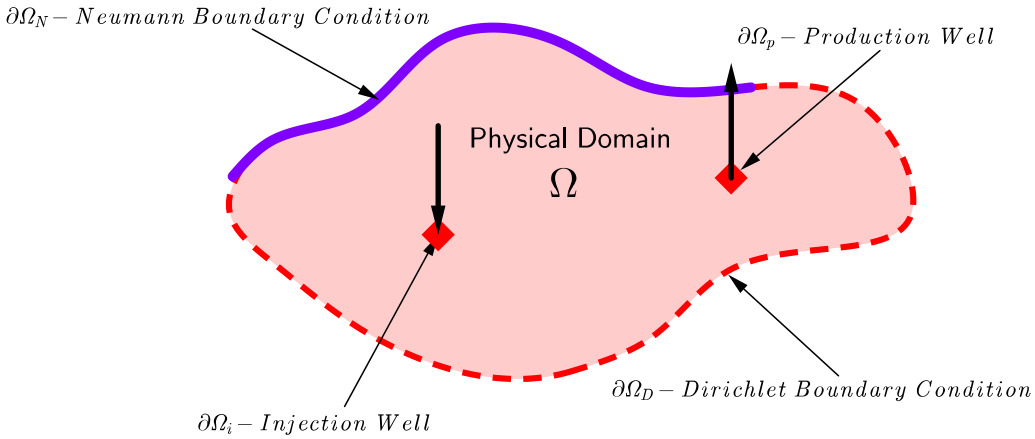
$$\vec{\nabla}\cdot(\underset{\sim}{K}\vec{\nabla}p) = Q \tag{2.10}$$

2.3.2 Saturation Equation

The saturation equation is a hyperbolic non-linear PDE that describes how one of the phases is transported throughout the porous medium. It can also be derived by combining and manipulating equations (2.1) and (2.4) for the water phase, leading to:

$$\phi\frac{\partial S_w}{\partial t} = -\vec{\nabla}\cdot\vec{F}(S_w) + Q_w \quad \text{for} \quad \Omega\times[0,t] \tag{2.11}$$

Figure 1 – Types of boundary conditions.



Source: Author.

in which $\vec{F}(S_w)$ is given by:

$$\vec{F}(S_w) = f_w(S_w)\vec{v} \quad \text{with} \quad f_w = \lambda_w/\lambda \tag{2.12}$$

The fractional flux $f_w(S_w)$ expresses the fraction of water being transported with the total flow. Notice that as the fractional flux depends on the saturation and it represents a non-linear term of the equation.

2.3.3 Initial and Boundary Conditions

In order to make the mathematical model described by equations 2.6 and 2.11 well-posed, a combination of proper initial and boundary conditions (BC) must be defined. Besides the classic boundary condition, there must exist conditions to represent the interaction of wells and the porous medium. The flow rate in injections wells q_i is equivalent to a Neumann condition, while the controlled pressure on the Production Well q_p is the Dirichlet condition.

The most common boundary conditions for the pressure equation are (CONTRERAS, 2012; BARBOSA, 2017; CARVALHO, 2005):

$$\begin{aligned} p(\vec{x},t) &= g_D \quad \text{in} \quad \partial\Omega_D \\ \vec{v}\cdot\vec{n} &= g_N \quad \text{in} \quad \partial\Omega_N \\ p(\vec{x},t) &= p_{\text{production}} \quad \text{in} \quad \partial\Omega_p \\ \vec{v}\cdot\vec{n} &= Q_{\text{injection}} \quad \text{in} \quad \partial\Omega_i \end{aligned} \tag{2.13}$$

The first two of these conditions on equation 2.13 are respectively Dirichlet (g_D) and Neumann (g_N) BC or prescribed pressure and prescribed flux defined on the contour of the domain. The following two are the same Dirichlet and Neumann conditions but defined on the production ($p_{\text{production}}$) and injections wells $Q_{\text{injection}}$.

For the pressure equation boundary condition need to be defined on the whole contour of the domain $\partial\Omega$. The BC becomes a disjoint union of the different boundary conditions adequately chosen for each problem as shown in Figure 1, and:

$$\partial\Omega = \partial\Omega_i + \partial\Omega_p + \partial\Omega_D + \partial\Omega_N \tag{2.14}$$

As for the BC of the saturation equation it is required to define the water saturation in the injection wells ($\overline{S}_w$) and the initial conditions are set by prescribing the water saturation ($\overline{S}_w^0$) on the domain Ω at the initial time step $t = t_o$.

$$\begin{array}{lll} S_w(\vec{x},t) = \overline{S}_w & \text{in} & \partial\Omega \times [0,t] \\ S_w(\vec{x},0) = \overline{S}_w^0 & \text{in} & \Omega \end{array} \tag{2.15}$$

3 NUMERICAL FORMULATION

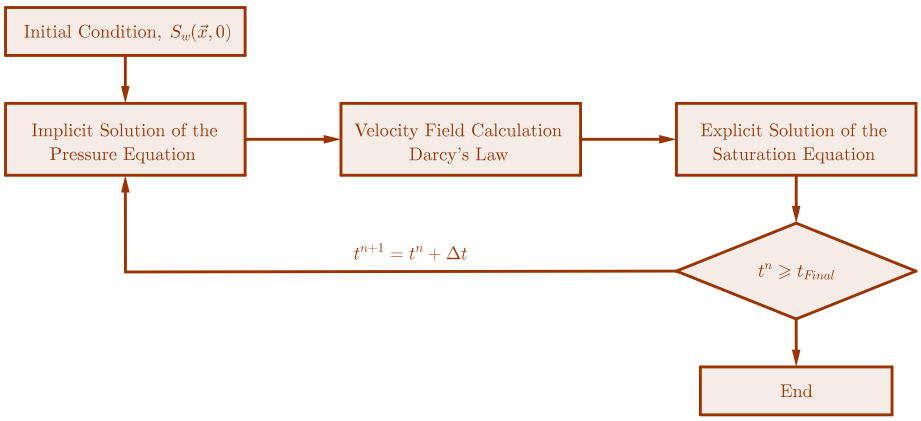
Several methods have been devised to solve systems of partial differential equations. Among them, the Finite Volume Method (FVM) stands out as a scheme that provides conservative solutions. This means that no fluid can be created or destroyed inside this system. Moreover, the Finite Volume Method (FVM) has its origins in a physical interpretation of the flow problem (SOUZA, 2015; CARVALHO, 2005; CONTRERAS et al., 2016).

In this chapter, we present the basic numerical strategies used to solve the PDEs presented in last chapters. Throughout this thesis we employ a cell-centered Finite Volume approximation.

3.1 IMPES strategy

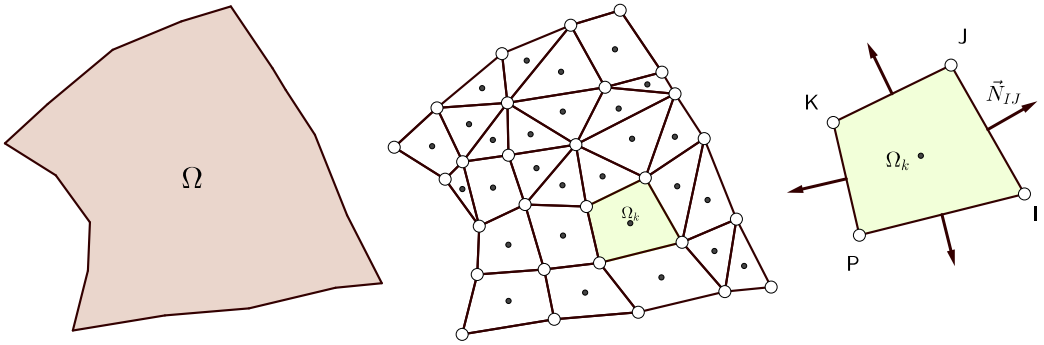
In the following sections, we use the Finite Volume Method to derive a discrete form of the pressure and saturation equations. We apply the IMPES(Implicit Pressure Explicit Saturation) strategy in which the pressure equation implicitly solved and saturation equation explicitly. See Figure 2 for details. First, the total mobility λ_T is calculated using an initial saturation field, then the pressure equation is implicitly solved; the velocity fields is explicitly calculated using Darcy's law and finally a new Saturation field is also explicitly computed. If a time-related condition is met, such as the simulation has reached the desired time, this process stops, if not, the new saturation field is used as input and this process repeats itself until this condition is met.

Figure 2 – IMPES Flowchart.



Source: Author.

Figure 3 – Discrete domain and general control volume Ω_k.



Source: Author.

3.2 Discrete Form of the Pressure Equation

According to the Finite Volume Method, the approximate solution of the pressure equation is obtained by integrating equation (2.6) throughout the domain Ω. This leaves us with:

$$\int_{\Omega} \vec{\nabla} \cdot \vec{v} dV = \int_{\Omega} Q dV \tag{3.1}$$

In order to obtain a discrete form of equation 3.1, the domain Ω is discretized in N_k general control volumes (CVs) as illustrated in the figure 3. This allows to rewrite equation (3.1) as the sum of the integrals of all N_k CV:

$$\sum_{1}^{N_k} \int_{\Omega_k} \vec{\nabla} \cdot \vec{v} dV = \sum_{1}^{N_k} \int_{\Omega_k} Q dV \tag{3.2}$$

Therefore, equation (3.1) can also be written for a single general control volume Ω_k as:

$$\int_{\Omega_k} \vec{\nabla} \cdot \vec{v} dV = \int_{\Omega_k} Q dV \tag{3.3}$$

By using the Divergence Theorem on the left hand side of the equation (3.3), we obtain:

$$\int_{\partial\Omega_k} \vec{v} \cdot \vec{n} dA = \int_{\Omega_k} Q dV \tag{3.4}$$

where $\partial\Omega_k$ is the surface of the control volume Ω_k and $\vec{n}$ the unitary vector normal to $\partial\Omega_k$.

These two integrals of equation (3.4) can be approximated using the Mean Value

Theorem as:

$$\begin{aligned} \int_{\partial\Omega_k} \vec{v}\cdot\vec{n}dA &= \sum_{IJ\in\partial\Omega_k} \vec{v}_{IJ}\cdot\vec{N}_{IJ} \\ \int_{\Omega_k} QdV &= \bar{Q}_k\Omega_k \end{aligned} \tag{3.5}$$

where $\vec{v}_{IJ}$ stands for the mean velocity approximated on an arbitrary linear surface formed by vertices I and J, $\vec{N}_{IJ}$ stands for the normal area vector for a random surface (or an edge in 2d) IJ and $\bar{Q}_k$ is the mean value of the source/sink term.

Then, equation (3.4) is rewritten as:

$$\sum_{IJ\in\partial\Omega_k} \vec{v}_{IJ}\cdot\vec{N}_{IJ} = \bar{Q}_k\Omega_k \tag{3.6}$$

where IJ stands for one edge of all edges that comprised the surface of a control volume $\partial\Omega_k$.

Notice that equation (3.6) is an exact discrete approximation of equation (3.4). In order to obtain a fully discrete system of equations, we need a consistent approximation for the pressure gradient present in the Darcy's flow.

3.2.1 Mobility Approximation

An important step of the IMPES strategy is obtaining consistent approximations for the mobilities in the pressure equation. FRIIS and EVJE (2012), Souza (2015) suggests the total mobility projected on an edge IJ can be obtained by the average of the approximation of the mobility on each node of this edges (J and I). (See figure 4).

$$\lambda_{IJ} = \frac{\lambda_I + \lambda_J}{2} \tag{3.7}$$

where the nodal mobilities λ_I and λ_J are approximated by the arithmetic mean of the mobilities surrounding a given a node:

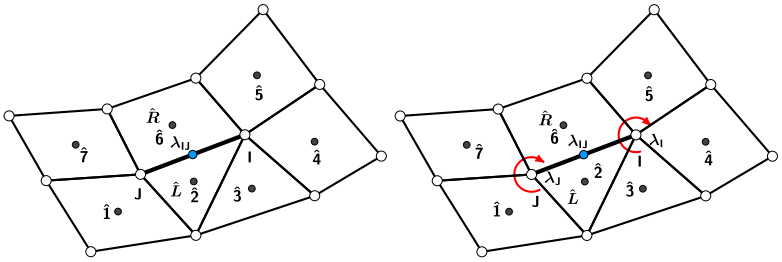
$$\lambda_i = \frac{\sum_{\hat{k}=1}^{n_i} \lambda_{\hat{k}}\Omega_{\hat{k}}}{\sum_{\hat{k}=1}^{n_i} \Omega_{\hat{k}}} \tag{3.8}$$

where $i = I, J$ and n_i is number of control volumes around node i.

3.2.2 FV Flux Approximation

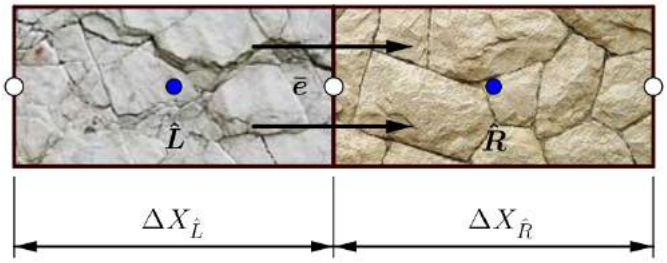
There are several possibilities to compute an approximate value for the flux at the surface of the CVs. Each choice represents a different formulation with advantages

Figure 4 – Mobility Approximation.



Source: Author.

Figure 5 – Unidimensional domain.



Source: Author.

and disadvantages (AAVATSMARK et al., 1998; CRUMPTON; SHAW; WARE, 1995; CONTRERAS, 2012; CONTRERAS et al., 2016; EDWARDS; ROGERS, 1998; EDWARDS; ZHENG, 2008; GAO; WU, 2010).

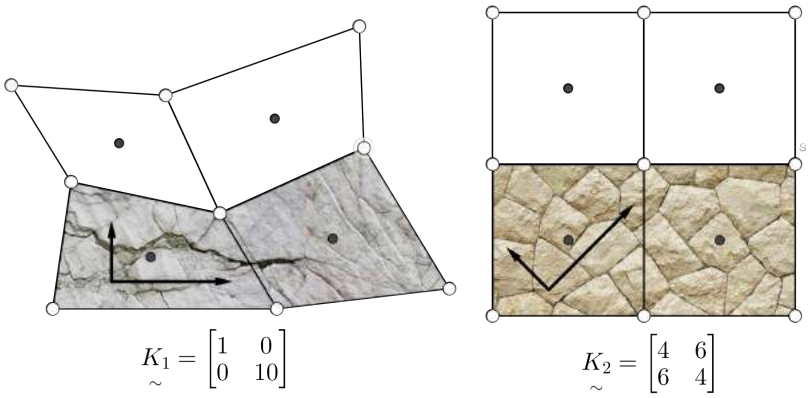
3.2.2.1 Two Point Flux Approximation - TPFA

The classical FVM approach for the discretization of Darcy's flow is a Two-Point Flux Approximation (TPFA) scheme. It is an accurate, robust and efficient method however it lacks support for full permeability tensors and unstructured grids (SOUZA, 2015).

In order to understand how TPFA is devised, let us consider a fragment of unidimensional domain discretized by two cell-centered CVs as illustrated on Figure 5. Let us consider a pressure gradient driving flow from the left-side $\hat{L}$ to the right-side control volume $\hat{R}$. Both volumes possess different permeabilities, turning the middle-edge permeability discontinuous. However, for a consistent Finite Volume Approximation, flux must be unique and continuous between any two adjacent volumes.

In this context, we derive the TPFA in three straightforward steps. First, we assume piecewise linear flux in each control volumes to approximate flux at the middle-

Figure 6 – TPFA consistency errors.



Source: Author.

edge:

$$\begin{aligned} (\vec{v}_{\bar{e}} \cdot \vec{N}_{\bar{e}})_{\hat{L}} &= -K_{\hat{L}} A \left(\frac{p_{\bar{e}} - p_{\hat{L}}}{\frac{\Delta X_{\hat{L}}}{2}} \right) \\ (\vec{v}_{\bar{e}} \cdot \vec{N}_{\bar{e}})_{\hat{R}} &= -K_{\hat{R}} A \left(\frac{p_{\bar{e}} - p_{\bar{R}}}{\frac{\Delta X_{\hat{R}}}{2}} \right) \end{aligned} \tag{3.9}$$

where $p_{\bar{e}}$ is an auxiliary pressure at middle-edge and A the area of the face.

Second, we impose continuity by assuming that the flux is equal on both sides. As a consequence we find an unique value for $p_{\bar{e}}$:

$$\begin{aligned} (\vec{v}_{\bar{e}} \cdot \vec{N}_{\bar{e}})_{\hat{L}} + (\vec{v}_{\bar{e}} \cdot \vec{N}_{\bar{e}})_{\hat{R}} &= 0 \\ -K_{\hat{L}} \left(\frac{p_{\bar{e}} - p_{\hat{L}}}{\frac{\Delta X_{\hat{L}}}{2}} \right) &= -K_{\hat{R}} \left(\frac{p_{\hat{R}} - p_{\bar{e}}}{\frac{\Delta X_{\hat{R}}}{2}} \right) \\ p_{\bar{e}} &= \frac{\frac{p_{\hat{L}} K_{\hat{L}}}{\Delta X_{\hat{L}}} + \frac{p_{\hat{R}} K_{\hat{R}}}{\Delta X_{\hat{R}}}}{\frac{K_{\hat{L}}}{\Delta X_{\hat{L}}} + \frac{K_{\hat{R}}}{\Delta X_{\hat{R}}}} \end{aligned} \tag{3.10}$$

Finally, we derive a single continuous flux on the middle-edge by inserting $p_{\bar{e}}$ back into any of the equations (3.9):

$$(\vec{v}_{\bar{e}} \cdot \vec{N}_{\bar{e}}) = -\frac{2 K_{\hat{L}} K_{\hat{R}}}{K_{\hat{L}} \Delta X_{\hat{R}} + K_{\hat{R}} \Delta X_{\hat{L}}} A (p_{\hat{R}} - p_{\hat{L}}) \tag{3.11}$$

This approximation has been proven to be accurate, efficient and robust for multiphase-flow simulation on isotropic media and on general k-orthogonal grids, i.e., grids where the surfaces of the cells are aligned with the principal directions of the permeability tensor (AAVATSMARK et al., 1998). In this approximation the flux across

a control surface only depends on the pressure of the volume on the left and right of this surface, which means the contribution of cross fluxes is not properly computed (SOUZA, 2015). Figure 6 shows situations where this approximation may lead to inconsistent flux approximation. The media depicted on the left figure shows an anisotropic media on with a diagonal permeability tensor K_1. In this case the projection of the permeability tensor on an edge of the grid may result in a full permeability tensors which induces cross flux that in turn is not taken in account on the flux approximation of Equation 3.11. Moreover, the media on the right figure possess a full permeability tensor (eg.: K_2) on a k-orthogonal grid. Note that this tensor induces a cross flux 1.5 greater than flux on the main directions. Once more, as Equation 3.11 neglects cross terms of the flux we have an inconsistent approximation (SOUZA, 2015; CONTRERAS et al., 2016). Hence, simulations on non isotropic media or on no k-orthogonal grids produces inconsistent approximated fluxes on the control surfaces which leads to first order errors $O(1)$ that do not disappear as the mesh is refined (EDWARDS; ZHENG, 2008).

3.2.2.2 Non-Orthodox FV Flux Approximation

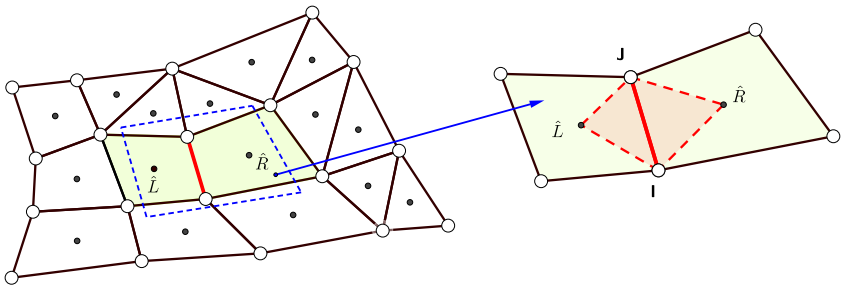
A lot of effort has been put to devise schemes capable of dealing with the limitations of the classical TPFA method. Crumpton, Shaw and Ware (1995) were the first to develop a control volume distributed scheme employing an approximation of the flux using multiple points. Edwards and Rogers (1998) and Aavatsmark et al. (1998), independently, generalized this formulation and devised control-volume distributed multi point flux approximation (CVD-MPFA) schemes to general unstructured grids. These strategies were successful in overcoming limitations that bounded TPFA to k-orthogonal grids. However, for high anisotropic media on highly distorted meshes, they may fail to converge (EDWARDS; ZHENG, 2008; CHEN et al., 2008). Hence, several authors have proposed more robust numerical schemes in order to deal with this limitation (LIPNIKOV et al., 2007; EDWARDS; ZHENG, 2008; SHENG; YUAN, 2011; CHEN et al., 2008; GAO; WU, 2010).

3.2.2.2.1 Multi Point Flux Approximation with a Diamond stencil - MPFA-D

In this thesis, we use the Multi Point Flux Approximation with a Diamond stencil (MPFA-D) originally proposed by Gao and Wu (2010) and brought to the petroleum reservoir context by Contreras et al. (2016). This approximation is a robust cell-centered scheme which ensures linearity preserving property. Therefore, MPFA-D can be seen as a generalization of the equation (3.11).

Before deriving MPFA-D flux, we shall state a lemma (GAO; WU, 2010; CONTRERAS et al., 2016; CONTRERAS, 2012) throughly used throughout this thesis:

Figure 7 – Zoom on the edge of two adjacent volumes inside the discrete domain.



Source: Author.

Lemma 1. *Let $\triangle OPQ$ be a triangle with vertices O, P, Q ordered counterclockwise. For a general pressure p defined on $\triangle OPQ$, we have:*

$$\vec{\nabla} p \simeq \frac{p_q - p_p}{|PQ|^2}\overrightarrow{PQ} + \frac{\mathcal{R}\overrightarrow{PQ}}{|PQ|^2}[(p_p - p_o)\cot\angle PQO + (p_q - p_o)\cot\angle OPQ] \tag{3.12}$$

where $\mathcal{R}$ is a linear operator which rotates a vector 90° clockwise. For a two-dimensional space, it can be defined as:

$$\mathcal{R} = \begin{bmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{bmatrix}, \quad with \quad \theta = 90^o \tag{3.13}$$

Check (CONTRERAS, 2012) for a proof of Lemma 1.

Let IJ be a general edge inside the discretization of a physical domain Ω as shown in Figure 7. Regardless the shape of any adjacent volumes inside this domain, two auxiliary triangular CVs can be formed by connecting the nodes I and J edge to the centroids of the volumes on the left $\hat{L}$ and on the right $\hat{R}$ of IJ. In a three-dimensional space, theses triangles become diamond-shaped tetrahedrons which give name to this scheme.

Assuming that $h_{IJ}^{\hat{L}}$ and $h_{IJ}^{\hat{R}}$ stand for the height of left and right triangles, $\vec{N}_{IJ} = \mathcal{R}\vec{IJ}$, $\vec{N}_{JI} = \mathcal{R}\vec{JI}$ and $\vec{N}_{IJ} + \vec{N}_{JI} = 0$, let us start deriving the MPFA-D flux expression. Figure 8 illustrates better these triangles as well as these parameters.

Let us begin by applying equation (3.12) on the left triangle $\triangle\hat{L}JI$.

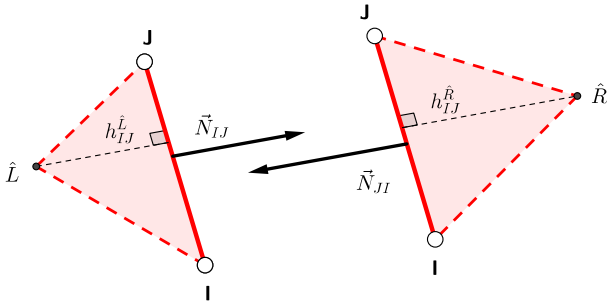
$$\vec{\nabla} p \simeq \frac{p_I - p_J}{|IJ|^2}\overrightarrow{IJ} + \frac{\mathcal{R}\overrightarrow{IJ}}{|IJ|^2}[(p_I - p_{\hat{L}})\cot\angle IJ\hat{L} + (p_J - p_{\hat{L}})\cot\angle\hat{L}IJ] \tag{3.14}$$

where $p_k = p(\vec{x})$ is a pressure defined on the node a k on the $\vec{x} = (x, y)$ coordinate.

We assume the full permeability tensor $\underline{K}$ projected on IJ is a linear combination of the tangential and normal permeabilities, as described bellow:

$$K_{IJ_{(i)}} = K^t_{IJ_{(i)}}\overrightarrow{IJ} + K^n_{IJ_{(i)}}\vec{N}_{IJ} \tag{3.15}$$

Figure 8 – Left and Right triangles.



Source: Author.

where (i) is equal to $\hat{L}$ or $\hat{R}$, so $K_{IJ_{(\hat{L})}}$ and $K_{IJ_{(\hat{R})}}$ stand for the projection of the permeability tensor of the left and right control volumes on the face IJ.

Let us now find expressions for the tangential K_{IJ}^t and normal permeability K_{IJ}^n projected on the edge IJ. By manipulating equation (3.15) we are able to obtain the following expression for normal and tangential permeabilities:

$$K_{IJ_{(i)}}^n = \frac{\vec{N}_{IJ}^T \underline{K}_{(i)} \vec{N}_{IJ}}{|IJ|^2} \tag{3.16}$$

$$K_{IJ_{(i)}}^t = \frac{\vec{N}_{IJ}^T \underline{K}_{(i)} \overrightarrow{IJ}}{|IJ|^2} \tag{3.17}$$

Note that, from Figure 8, we can also obtain these simple geometric expressions:

$$\cot \angle IJ\hat{L} = \frac{\overrightarrow{J\hat{L}} \cdot \overrightarrow{JI}}{|IJ| h_{IJ}^{\hat{L}}} \tag{3.18}$$

$$\cot \angle \hat{L}IJ = \frac{\overrightarrow{I\hat{L}} \cdot \overrightarrow{IJ}}{|IJ| h_{IJ}^{\hat{L}}} \tag{3.19}$$

We can derive an expression for the left flux by using equation (3.14) in (2.8) and applying on both sides the inner product of $\vec{N}_{IJ}$. See triangle $\triangle IJ\hat{L}$ on Figure (8). It follows:

$$\vec{v}_{IJ}^{\hat{L}} \cdot \vec{N}_{IJ} \simeq -\lambda_{IJ}[K_{IJ_{(\hat{L})}}^n \left((p_I - p_{\hat{L}}) \cot \angle IJ\hat{L} + (p_J - p_{\hat{L}}) \cot \angle \hat{L}IJ \right) - K_{IJ_{(\hat{L})}}^t (p_I - p_J)] \tag{3.20}$$

By using (3.16) to (3.19) and manipulating the resulting expression we obtain:

$$\frac{h_{IJ}^{\hat{L}}}{\lambda_{IJ} K_{IJ_{(\hat{L})}}^n} \vec{v}_{IJ}^{\hat{L}} \cdot \vec{N}_{IJ} \simeq -\frac{1}{|IJ|} \left((p_I - p_{\hat{L}}) \frac{\overrightarrow{J\hat{L}} \cdot \overrightarrow{JI}}{|IJ|} + (p_J - p_{\hat{L}}) \frac{\overrightarrow{I\hat{L}} \cdot \overrightarrow{IJ}}{|IJ|} \right) - (p_I - p_J) h_{IJ}^{\hat{L}} \frac{K_{IJ_{(L)}}^t}{K_{IJ_{(L)}}^n} \tag{3.21}$$

Similarly, we can redo the same expression on the triangle formed by the right control volume $\triangle I\hat{R}J$:

$$\frac{h_{IJ}^{\hat{R}}}{\lambda_{IJ}K_{IJ(\hat{R})}^{n}}\vec{v}_{IJ}^{\hat{R}}\cdot\vec{N}_{IJ}\simeq-\frac{1}{|IJ|}\left((p_J-p_{\hat{R}})\frac{\overrightarrow{J\hat{R}}\cdot\overrightarrow{JI}}{|IJ|}+(p_I-p_{\hat{R}})\frac{\overrightarrow{I\hat{R}}\cdot\overrightarrow{IJ}}{|IJ|}\right)-(p_I-p_J)h_{IJ}^{\hat{R}}\frac{K_{IJ(R)}^{t}}{K_{IJ(R)}^{n}} \tag{3.22}$$

As we have previously stated, for a consistent finite volume approximation flux must be continuous and unique across IJ, this means:

$$\vec{v}_{IJ}^{\hat{L}}\cdot\vec{N}_{IJ}+\vec{v}_{IJ}^{\hat{R}}\cdot\vec{N}_{JI}=0 \tag{3.23}$$

In which:

$$-\vec{v}_{IJ}^{\hat{L}}\cdot\vec{N}_{IJ}=\vec{v}_{IJ}^{\hat{R}}\cdot\vec{N}_{JI}=\vec{v}_{IJ}\cdot\vec{N}_{IJ} \tag{3.24}$$

Finally, we can use (3.23) in (3.22), take the average of the resulting expression and equation (3.21) to derive an expression for a unique and continuous flux on IJ:

$$\vec{v}_{IJ}\cdot\vec{N}_{IJ}\simeq\tau_{IJ}[p_{\hat{R}}-p_{\hat{L}}-\upsilon_{IJ}(p_J-p_I)] \tag{3.25}$$

where the scalar transmissibility τ_{IJ}, and the non-dimensional tangential parameter υ_{IJ} are defined as:

$$\tau_{IJ}=-\lambda_{IJ}|IJ|\frac{K_{IJ(\hat{L})}^{n}K_{IJ(\hat{R})}^{n}}{K_{IJ(\hat{L})}^{n}h_{IJ}^{\hat{R}}+K_{IJ(\hat{R})}^{n}h_{IJ}^{\hat{L}}} \tag{3.26}$$

$$\upsilon_{IJ}=\frac{\overrightarrow{IJ}\cdot\overrightarrow{\hat{L}\hat{R}}}{|IJ|^2}-\frac{1}{|IJ|}\left(\frac{K_{IJ(\hat{L})}^{t}}{K_{IJ(\hat{L})}^{n}}h_{IJ}^{\hat{L}}+\frac{K_{IJ(\hat{R})}^{t}}{K_{IJ(\hat{R})}^{n}}h_{IJ}^{\hat{R}}\right) \tag{3.27}$$

3.2.2.2.2 Boundary Condition Flux Treatment

For an adequate treatment of the boundary conditions each case given in (2.13) must be treated individually and accordingly. A flux expression for edges subjected to Dirichlet Boundary Conditions can be derived by using $g_D(I)$ and $g_D(J)$, prescribed pressures, in equation (3.20). It follows:

$$\vec{v}_{IJ}\cdot\vec{N}_{IJ}\simeq-\frac{\lambda_{IJ}K_{IJ}^{n}}{h_{IJ}^{\hat{L}}|IJ|}\left[(\overrightarrow{J\hat{L}}\cdot\overrightarrow{JI})g_D(I)+\overrightarrow{I\hat{L}}\cdot\overrightarrow{IJ})g_D(J)-p_{\hat{L}}|IL|^2\right]-K_{IJ}^{t}(g_D(J)-g_D(I)) \tag{3.28}$$

Similarly, for edges subjected to Neumann boundary conditions we have:

$$\vec{v}_{IJ}\cdot\vec{N}_{IJ}=g_N|IJ| \tag{3.29}$$

where g_N stands for the velocity normal to edge IJ.

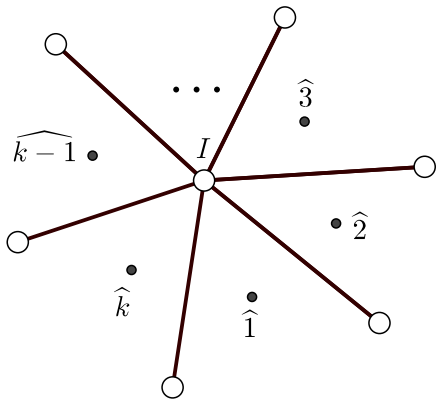
3.2.2.2.3 Linearity-Preserving Explicit Weighted Derivation

In cell-centered finite volume approximation all degrees of freedom are projected on the cell centroid. However, the expression (3.25) relies on two auxiliary nodes I and J. (See Figure 9). A simple solution to this problem is to define the pressure in such nodes as a linear combination of the cell pressures around theses nodes:

$$p_I = \sum_{k=1}^{n(I)} w_k p_k \tag{3.30}$$

where $n(I)$ is the number of volumes around I and w_k is the weighted attributed to pressure p_k.

Figure 9 – Auxiliary nodes can be written as linear combination of its surrounding neighbors.



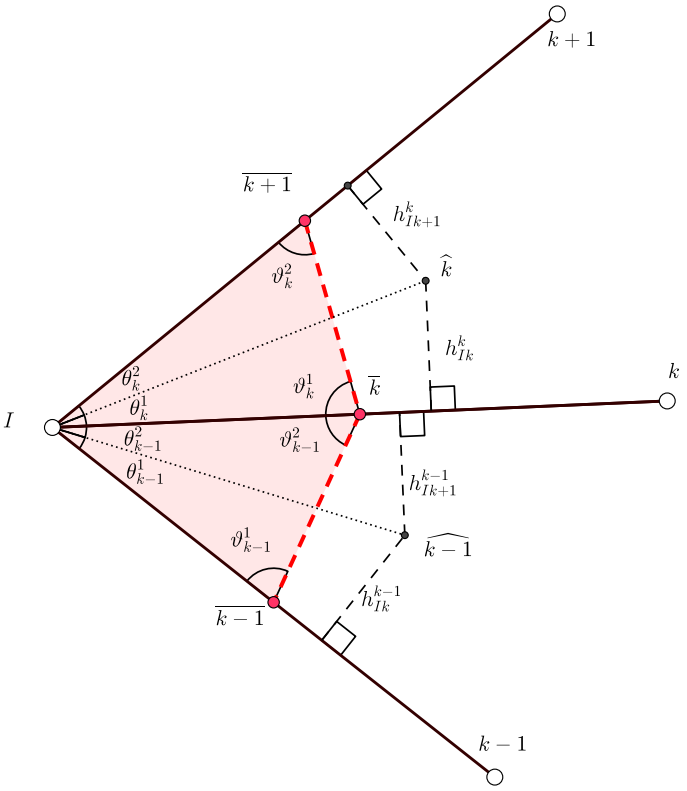
Source: Author.

Many authors have proposed ways to interpolate this pressures. Gao and Wu (2010) have developed two Linearity-Preserving Explicit Weighted (LPEW) interpolation methods. Both are explicit, since no system of equations are locally solved in the process, and robust for anisotropic and heterogeneous media (CONTRERAS et al., 2016). In this work we use the interpolation method LPEW-2.

The process of deriving LPEW-2 is very straightforward. Figure 10 illustrates all geometric entities between two adjacent volumes $\widehat{k}$ and $\widehat{k-1}$ used throughout the LPEW-2 deriving process. First we need to construct an auxiliary control volume by connecting each middle edge around the nodes to be interpolated (see figure 11a). Note that k triangles $\triangle I\overline{k}\,\overline{k+1}$, where $\overline{k+n} = \overline{n}$ and $k = n(I)$, are formed in this process.

First, we need calculate a discrete and continuous Darcy's flux on the middle edges $I\overline{k}$. We do this by applying lemma 1 on the triangle $\triangle I\overline{k}\,\overline{k+1}$ and on the triangle

Figure 10 – Sketch of the geometrical entities used in the LPEW-2 deriving process.



Source: Author.

$\triangle I\overline{k-1}\,\overline{k}$ (see Figure 11b). This lead us to the following expressions:

$$\vec{v}^{\hat{k}}_{I\overline{k}} \cdot \vec{N}_{I\overline{k}} \simeq K^n_{\hat{k},1}\eta^1_k(p_{\hat{k}} - p_I) - (K^t_{\hat{k},1} - \cot\vartheta_{\hat{k},1}K^n_{\hat{k},1})(p_{\overline{k}} - p_I) \tag{3.31}$$

$$\vec{v}^{\widehat{k-1}}_{I\overline{k}} \cdot \vec{N}_{\overline{k}} \simeq K^n_{\widehat{k-1},1}\eta^2_{\widehat{k-1}}(p_{\widehat{k-1}} - p_I) - (K^t_{\widehat{k-1},2} - \cot\vartheta_{\widehat{k-1},2}K^n_{\widehat{k-1},2})(p_{\overline{k}} - p_I) \tag{3.32}$$

where:

$$\eta^1_k = \frac{|I\overline{k}|}{h^k_k} \tag{3.33}$$

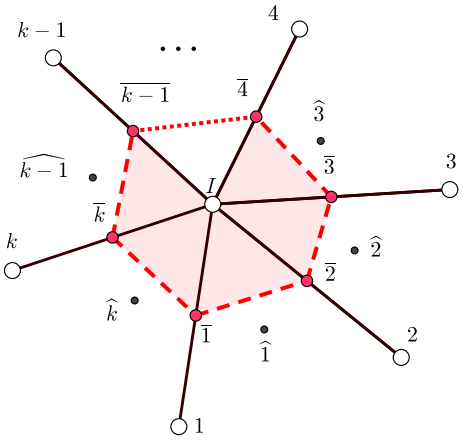
$$\eta^2_k = \frac{|I\overline{k+1}|}{h^k_{k+1}} \tag{3.34}$$

$$K^n_{\hat{\alpha}i} = \frac{\vec{N}^T_{I\bar{t}}\underline{K}_{\hat{\alpha}}\vec{N}_{I\bar{t}}}{|I\bar{t}|^2} \tag{3.35}$$

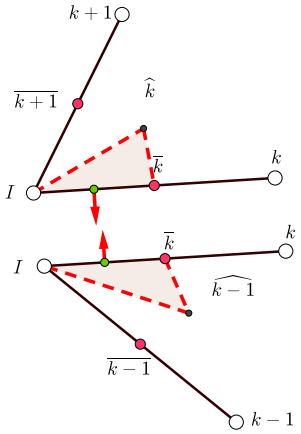
$$K^t_{\hat{\alpha}i} = \frac{\vec{N}^T_{I\bar{t}}\underline{K}_{\hat{\alpha}}\vec{I\bar{t}}}{|I\bar{t}|^2} \tag{3.36}$$

and where $\hat{\alpha} = 1, 2, ..., n(I)$, $i = 1, 2$, $t = \alpha + i - 1$.

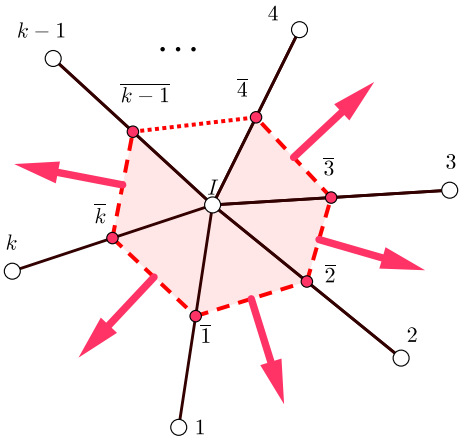
Figure 11 – LPEW 2 Derivation adapted from (GAO; WU, 2010).



(a) Auxiliary Control Volume around node I created by connecting middle edges.



(b) Imposing flux conservation on each the middle-edges of the auxiliary CV.



(c) Imposing flux conservation in the auxiliary control volume.

Source: Author.

The mass conservation equation written for the middle edge $I\overline{k}$ gives us:

$$\vec{v}_{I\overline{k}}^{\hat{k}} \cdot \vec{N}_{I\overline{k}} + \vec{v}_{I\overline{k}}^{\widehat{k-1}} \cdot \vec{N}_{\overline{k}} = 0 \tag{3.37}$$

After some algebraic manipulation, we obtain the following equation:

$$(p_{\hat{k}} - p_I) = \frac{K^n_{\hat{k},1}\eta^1_k(p_{\hat{k}} - p_I) + K^n_{\widehat{k-1},1}\eta^2_{\widehat{k-1}}(p_{\widehat{k-1}} - p_I)}{K^n_{\widehat{k-1},2}\cot\vartheta_{\widehat{k-1},2} + K^n_{\hat{k},1}\cot\vartheta_{\hat{k},1} - K^t_{\widehat{k-1},2} + K^t_{\hat{k},1}} \tag{3.38}$$

Flux of the surface of the auxiliary control volume $\overline{k-1\,k}$ is calculated analogously. It follows that:

$$\vec{v}_{\overline{k\,k+1}} \cdot \vec{N}_{\overline{k\,k+1}} \simeq \left(\overline{K}^t_{\hat{k}} - \overline{K}^n_{\hat{k}}\cot(\vartheta^2_{\hat{k}})\right)(p_{\overline{k}} - p_I) - \left(\overline{K}^t_{\hat{k}} + \overline{K}^n_{\hat{k}}\cot(\vartheta^1_{\hat{k}})\right)(p_{\overline{k+1}} - p_I) \tag{3.39}$$

where $\vartheta^1_{\hat{k}} = \angle I\overline{k}\,\overline{k+1}$ and $\vartheta^2_{\hat{k}} = \angle\overline{k}\,\overline{k+1}I$

$$K^n_{\hat{\alpha}} = \frac{\vec{N}^T_{\overline{\alpha}\,\overline{\alpha+1}}\underset{\sim}{K}_{\alpha}\vec{N}^T_{\overline{\alpha}\,\overline{\alpha+1}}}{|\overline{\alpha}\,\overline{\alpha+1}|^2} \tag{3.40}$$

$$K^t_{\hat{\alpha}} = \frac{\vec{N}^T_{\overline{\alpha}\,\overline{\alpha+1}}\underset{\sim}{K}_{\alpha}\overrightarrow{\overline{\alpha}\,\overline{\alpha+1}}}{|\overline{\alpha}\,\overline{\alpha+1}|^2} \tag{3.41}$$

where $\alpha = 1, 2, ..., n(i)$.

We also impose flux conservation on the boundaries of auxiliary control volume (see Figure 11c):

$$\sum_{k=1}^{n(I)} v_{\overline{k}\,\overline{k+1}} \cdot \vec{N}^t_{\overline{k}\,\overline{k+1}} = 0 \tag{3.42}$$

By using (3.40) and (3.41) in (3.39) and (3.42) and after substituting (3.38) on the resulting expression, we derive the LPEW-2 weights as:

$$w_k = \frac{\psi_k}{\sum_{k=1}^{n(I)}\psi_k} \tag{3.43}$$

where

$$\psi = \overline{K}^n_{\hat{k},1}\eta^1_k\xi_k + \overline{K}^n_{\hat{k},2}\eta^2_k\xi_{k+1} \tag{3.44}$$

and where:

$$\xi_k = \frac{\overline{K}^t_{\hat{k-1}} - \overline{K}^t_{\hat{k}} + \overline{K}^n_{\hat{k-1}}\cot\vartheta^1_{\hat{k-1}} + \overline{K}^n_{\hat{k}}\cot\vartheta^2_{\hat{k}}}{K^{t,1}_k - K^{t,2}_{k-1} + K^{n,2}_{k-1}\cot\theta^2_{k-1}K^{n,1}_k\cot\theta^1_k} \tag{3.45}$$

Nodes lying on the boundary of the computational domain must be treated accordingly. For nodes subjected to Dirichlet BC one can simply set the values of node I and J to the value of the prescribed pressure. In cases of nodes lying in a Neumamm BC, weights must be calculated taking in account incoming and out-coming fluxes. Essentially the deriving process is the same and it gives the following results:

$$\xi_k = \begin{cases} \frac{\bar{K}^n_1\cot\vartheta^2_1 - \bar{K}^t_1}{K^n_1\cot\vartheta^1_1 - K^t_1}, & k = 1; \\ \text{Use equation (3.45)}, & 2 \leqslant k \leqslant n(I); \\ \frac{\bar{K}^n_{n(I)}\cot\vartheta^2_{n(I)} + \bar{K}^t_{n(I)}}{K^n_{n(I)}\cot\vartheta^1_{n(I)} - K^t_{n(I)}}, & k = n(I)+1; \end{cases} \tag{3.46}$$

where k is counted counterclockwise from the edge 1 until the last edge around this node $n(I)+1$. Check (GAO; WU, 2010; CONTRERAS, 2012; CONTRERAS et al., 2016) for more detail.

3.3 Discrete Form of the Saturation Equation

In order to derive a discrete form of the saturation equation, we employ the same basic assumptions used to derive (3.6). We start by integrating equation (2.9) in the time interval t_o and t and in the domain Ω obtaining:

$$\int_{t_o}^{t}\int_{\Omega}\frac{\partial S_w}{\partial t}dVdt = -\int_{t_o}^{t}\int_{\Omega}\frac{1}{\phi}\vec{\nabla}\cdot\vec{F}(S_w)dVdt + \int_{t_o}^{t}\int_{\Omega}\frac{1}{\phi}Q_w dVdt \tag{3.47}$$

The space integral can be represented as a sum of integrals for all N_k control volumes in the domain Ω. After applying Gauss Theorem in the first integral on the RHS equation (3.47) is rewritten as:

$$\int_{t_o}^{t}\sum_{1}^{N_k}\int_{\Omega_{\hat{k}}}\frac{\partial S_w}{\partial t}dVdt = -\int_{t_o}^{t}\sum_{1}^{N_k}\frac{1}{\phi_{\hat{k}}}\int_{\partial\Omega_{\hat{k}}}\vec{F}(S_w)\vec{n}dsdt + \int_{t_o}^{t}\sum_{1}^{N_k}\int_{\Omega_{\hat{k}}}\frac{1}{\phi_{\hat{k}}}Q_w dVdt \tag{3.48}$$

where N_k is the number of control volumes that subdivide the physical domain Ω.

Equation (3.48) is also valid for a single control volume $\hat{k}$, i.e.:

$$\int_{t_o}^{t}\int_{\Omega_{\hat{k}}}\frac{\partial S_w}{\partial t}dVdt = -\frac{1}{\phi_{\hat{k}}}\int_{t_o}^{t}\int_{\partial\Omega_{\hat{k}}}\vec{F}(S_w)\vec{n}dsdt + \frac{1}{\phi_{\hat{k}}}\int_{t_o}^{t}\int_{\Omega_{\hat{k}}}Q_w dVdt \tag{3.49}$$

By using the mean value theorem, we get:

$$\frac{1}{\phi_{\hat{k}}}\int_{\partial\Omega_{\hat{k}}}\vec{F}(S_w)\vec{n}dsdt \simeq \frac{1}{\phi_{\hat{k}}}\sum_{IJ\in\Omega_{\hat{k}}}\vec{F}(S_w)_{IJ}\cdot\vec{N}_{IJ} \tag{3.50}$$

$$\frac{1}{\phi_{\hat{k}}}\int_{\Omega_{\hat{k}}}Q_w dV \simeq \frac{1}{\phi_{\hat{k}}}\overline{Q}_w \tag{3.51}$$

and

$$\int_{\Omega_{\hat{k}}}\frac{\partial S_w}{\partial t}dV \simeq \frac{\partial\overline{S_w}}{\partial t}\Omega_{\hat{k}} \tag{3.52}$$

where $\overline{Q}_w$ stand for a average sink/source term integrated on the referring CV.

Using (3.50) to (3.52) in (3.49), we obtain :

$$\int_{t_o}^{t} \frac{\partial S_w}{\partial t}\Big|_k dt = -\frac{1}{\phi_{\hat{k}}\Omega_k} \int_{t_o}^{t} \sum_{IJ \in \Omega_{\hat{k}}} \vec{F}(S_w)_{IJ} \cdot \vec{N}_{IJ} dt + \frac{\overline{Q}_w}{\phi_{\hat{k}}\Omega_{\hat{k}}} \tag{3.53}$$

which is the semi-discrete form of the saturation equation. Note that we have dropped the over bar notation for the saturation term on the LHS.

3.3.1 Temporal Discretization

The standard solution used by the oil and gas industry to discretize the time derivative term in the saturation equation for a two-phase flow simulation using the IMPES algorithm (see Figure 2) is the Forward Euler Method, a first order approximation.

Using the Forward Euler approximation to integrate the LHS of equation (3.53), we have:

$$\int_{t_o}^{t} \frac{\partial S_w}{\partial t} dt = S_{w,\hat{k}}^{n+1} - S_{w,\hat{k}}^{n} \tag{3.54}$$

where $S_{w,\hat{k}}^{n+1}$ and $S_{w,\hat{k}}^{n}$ are, respectively, the water saturation on $\hat{k}$ at the time level n and $n+1$.

By using (3.54) in (3.53) and manipulating we obtain a fully discrete saturation equation approximation:

$$S_{w,\hat{k}}^{n+1} = S_{w,\hat{k}}^{n} - \frac{\Delta t}{\phi_{\hat{k}}\Omega_{\hat{k}}} \sum_{IJ \in \Omega_{\hat{k}}} \vec{F}(S_w)_{IJ} \cdot \vec{N}_{IJ} + \frac{\Delta t \overline{Q}_w}{\phi_{\hat{k}}\Omega_{\hat{k}}} \tag{3.55}$$

where the hyperbolic term is $\vec{F}(S_w)_{IJ} = f_w\left(S_{w,\hat{k}}^{n}\right)\vec{v}_{IJ}$ and the time step is $\Delta t = t^{n+1} - t^n$.

Notice that we have derived an explicit approximation which only relies on current known information to calculate the saturation at next time step. Therefore, to ensure stability, Δt must satisfy Courant-Friedrichs-Lewy condition (CFL):

$$max_{IJ \in \Omega_{\hat{k}}} \left[\left(\frac{\Delta f_w(S_w)}{\Delta S_w} \right)_{IJ} \vec{v}_{IJ} \cdot \vec{N}_{IJ} \right] \frac{\Delta t}{V_{\hat{k}}} \leqslant \sigma \tag{3.56}$$

where the CFL condition is σ, and $\left(\frac{\Delta f_w(S_w)}{\Delta S_w}\right)_{IJ}$ is a discrete approximation of $(\frac{\partial f_w(S_w)}{\partial S_w})_{IJ}$ (SOUZA, 2015).

4 MULTISCALE METHOD

In this section we describe the modifications performed on the MsRSB (MOYNER; LIE, 2015) scheme to produce the Multiscale Control Volumed framework developed for solving the pressure equation in one-phase and two-phase flows on unstructured grids on all scales.

We start the chapter by presenting the classical Multiscale Finite Volume Method (MsFVM) (JENNY; LEE; TCHELEPI, 2003) and improved (JENNY; LEE; TCHELEPI, 2006) using the algebraic notation developed by Zhou and Tchelepi (2008) which allows us to describe general multiscale methods as a series of simple matrix operations by defining a restriction and a prolongation operator. General concepts used in Multiscale Finite Volume methods are also introduced, and finally we present a general algorithm for two-phase flow in porous media flow using a MsFV for the pressure equation only.

This chapter proceeds as we present the MsRSB along with the modifications used to couple a CVD MPFA-D pressure solver with a FOU saturation solver. Differently from the previous section, we also describe and discuss implementation details and algorithms developed for this thesis.

It is noteworthy that there are also Multiscale Finite Volume strategies available for the saturation equation (ZHOU, 2010) however in this thesis, we focus only on the studies of the pressure equation multiscale methodologies.

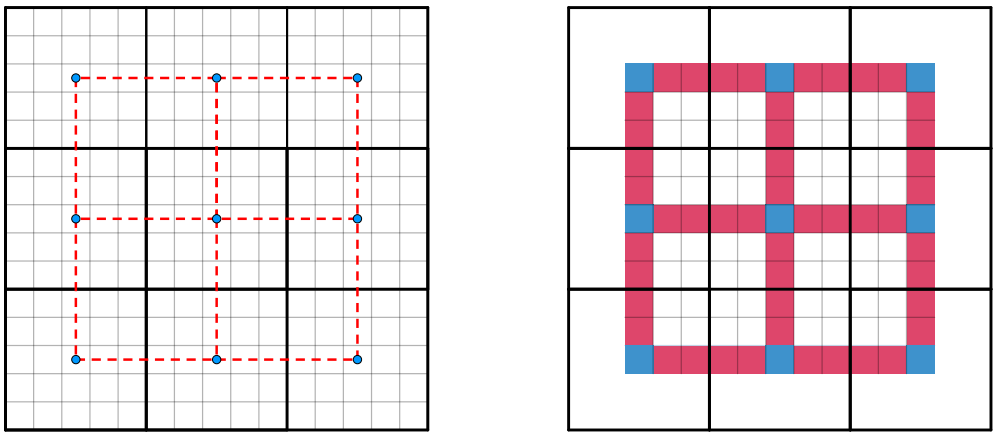
4.1 Multiscale Finite Volume Method

The formulation which is considered to be the standard Multiscale Finite Volume method was developed by Jenny, Lee and Tchelepi (2006) and it builds on his previous work (JENNY; LEE; TCHELEPI, 2003) and it suggest a fine-scale flux reconstruction step done by imposing Neumann boundary conditions on coarse scale volumes.

Additionally, we use the algebraic formulation developed by Zhou and Tchelepi (2008) to describe the scale transferring processing using two matrices operators: A Restriction Operator that restrains the influence of fine-scale cells onto single coarse volumes and Prolongation Operator used to map the influence of a coarse volume into fine-scale cells. This allows this method to be also used as an Upscaling technique. Moreover, as a Finite Volume Method, it inherits local mass conversation properties so important for a consistent approximation of the saturation field.

As most of the multiscale methods, including the MsFVM, were conceived to work strictly with k-orthogonal grids, all the illustrations and general concepts

Figure 12 – A description of coarse and fine meshes on the left. Center cells (blue), Edges Cells (red), Inner Dual (white).



Source: Author.

presented in this section have a structured mesh format. However, in the Section 4.3 we will revisit some of this concepts taking in account the generalizations required by unstructured meshes as we present the MsRSB.

4.1.1 Basic Multiscale Finite Volume Mesh Concepts

By definition all numerical schemes that work with scale-transferring process make use of some sort of auxiliary mesh. Classical Multiscale Methods such as MsFEM and MsMFEM rely simply on a higher-resolution grid, fine mesh, and on a lower-resolution grid, coarse mesh. In addition these the Finite Volume Multiscale Methods introduces a third grid used to ensure flux conservation on the fine scale. Figure 12 describes these threes types of meshes .

Fine Scale Mesh (Ω^f or Ω): This is the higher-resolution grid derived from the discretization of the physical domain. Traditional Multiscale Methods generally employ some kind of k-orthogonal grid discretization for this mesh. Figure 12 illustrates this mesh as the smaller gray squares volumes covering the domain on the left side.

Primal Coarse Mesh (Ω_c^p or Ω^p): This is a lower-resolution grid generated by clustering fine-scale volumes. In most traditional multiscale methods, it inherits the k-orthogonality of the underlying discretization, giving birth to simple forms of rectangular shapes. Figure 12 illustrates this mesh as thick bold black lines on the left side. Note that each coarse volume is composed of a finite amount of fine-scale volumes.

Dual Coarse Mesh (Ω_c^d or Ω^d): In the classical MsFVM and its variants, the Dual n other Coarse Mesh arises from the need to reimpose conservation on the boundaries of the coarse volumes. It is usually generated by connecting the centers of each adjacent coarse volume. The duality emerges as all the vertices of the primal coarse volumes are the centers of each dual coarse volumes. Figure 12 shows this process, the blue nodes (coarse volume centers) are connected with the other center forming dual volumes with the red-dashed boundary. On the right Figure 12, we zoom in to illustrate the composition of the dual volumes. Each dual coarse volume is comprised by boundary (red), inner (white) and coarse center volumes(blue).

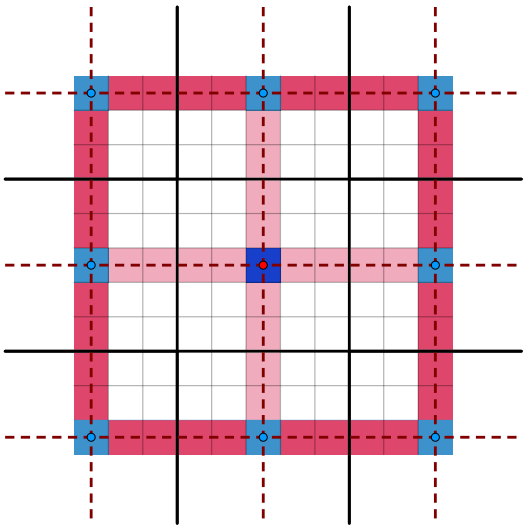
Primal Coarse Center (x^P): The fine-scale volume which will represent the center of a coarse volume. The classical MsFVM approach is to use the fine volume closest to the coarse cell centroid, however the recent work (BARBOSA, 2017) proposes shifting the primal coarse center of coarse volumes lying on the boundary of the domain. In these cases, the new center should be the fine-scale volume closest to the intersection between the boundary of volume and the physical domain. On the right, Figure 12, these volumes are represented in blue. The very definition of the Dual Coarse Mesh relies on how these center volumes are defined. Therefore, for unstructured grids a more robust algorithm needs to be applied. In the next section we describe a different algorithm (MOYNER; LIE, 2015) to be used for general unstructured coarse meshes.

Support of Region of Primal Coarse Volume: The concept of a support region of a coarse volume is intrinsically connected to the basis function idea. The mathematical definition of a support of a real-valued function is the subset of a domain containing non-zero elements only. Traditionally, MsFVM used this concept implicitly by using the Dual Coarse Mesh. See Figure 13. In these methods, the support region of Coarse Center (x^P) is comprised of dual coarse volumes(dashed red lines) surrounding x^P. Note that support of the dark blue cell center is the sum of the four dual cells around it. Not all the boundaries of the dual cells remain inside the support region, the light pink volumes located on the middle of the region do, however, volumes lying on the outside boundary do not. Different from the primal coarse (solid black lines) and fine mesh partition, the support region overlaps with support regions of different coarse centers to cover the whole domain. More details on Subsection 4.1.2.2.

Coarsening Ratio (C_r): The coarsening ratio or the upscaling ratio is defined as:

$$C_r = \frac{n_f}{n_c} \tag{4.1}$$

Figure 13 – Support Region



Source: Author.

where n_c and n_f are the numbers of coarse volumes and the number of fine volumes. It is worth pointing out the coarsening ratio has a direct impact on the quality of the multiscale solution.

4.1.2 MsFV Operators

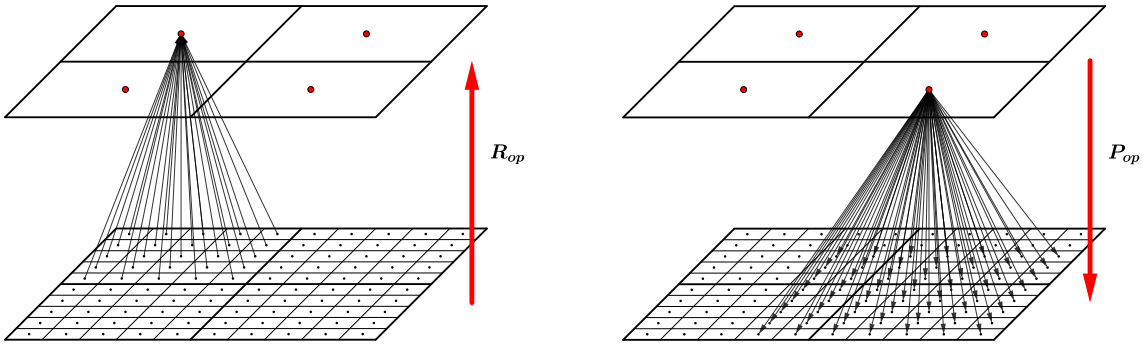
The very definition of Multiscale Methods arises from the need to exchange information in different grids. As a direct simulation on a higher-resolution mesh is not feasible, we transfer relevant information to a lower-resolution grid using a sort of restriction algorithm. After that, we solve the resulting coarse system and the solution is projected back onto the higher-resolution grid using a set of basis functions. In a simplified way, the lower resolution system works as an auxiliary basis with a considerable fewer degrees of freedom which enables the simulation, meanwhile the basis functions capture the coupling between these scales. Therefore, for a discrete system it is natural to define these scale-transferring operators in a matrix form.

4.1.2.1 Restriction Operator

The idea behind the Restriction Operator $\underline{R}_{op}$ is to create an operator capable of mapping information from the fine-scale (Ω_f) onto the coarse-scale (Ω_c^p). See Figure 14a. Moreover, for a consistent finite volume approximation, this operator must respect the flux conservation law and when written in matrix form the restriction operator must obey the following equation:

$$Q_c = \underline{R}_{op} Q_f \tag{4.2}$$

Figure 14 – Multiscale Operators



(a) Restriction Operator (b) Prolongation Operator

Source: Author.

where Q_c and Q_f are respectively a $n_c \times 1$ and $n_f \times 1$ source/sink term on the coarse and fine scale as a consequence $\underset{\sim}{R}_{op}$ must be a $n_c \times n_f$ matrix.

In essence, equation (4.2) states algebraically that when we apply the Restriction Operator to a fine-scale source/sink vector an equivalent and conservative vector on the coarse-scale space.

Most traditional MsFV Methods propose using the Volume Summation (CHEN; HOU, 2002; JENNY; LEE; TCHELEPI, 2006; ZHOU, 2010; MOYNER; LIE, 2015) as the Restriction Operator. It can be defined as:

$$(\underset{\sim}{R}_{op})_{ij} = \begin{cases} 1, & \Omega_i^f \in \Omega_j^c; \\ 0, & \text{otherwise} \end{cases} \quad \text{where } 1 \leqslant j \leqslant n_c \quad \text{and} \quad 1 \leqslant i \leqslant n_f \tag{4.3}$$

This operator literately works as a switch identifying which fine cells Ω_i^f influence the coarse volume Ω_j^c. As a consequence, j^{th} row of the operator is a $1 \times n_f$ boolean vector with 1 for all fine cells that belong to Ω_j^c and 0 otherwise.

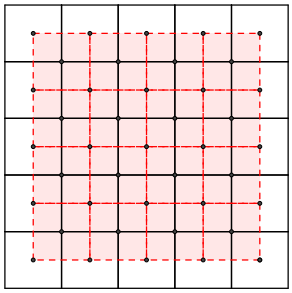
4.1.2.2 Prolongation Operator

On the other hand, the Prolongation Operator $\underset{\sim}{P}_{op}$ maps the influence of the coarse volumes onto the fine scale mesh. Concisely, the fine-scale space solution becomes a convex combination of the pressures on the coarse-scale space. See Figure 14a. The weighting function used to span the multiscale fine solution is called the basis function. By construction, the Prolongation Operator is comprised of all the basis functions in such way that:

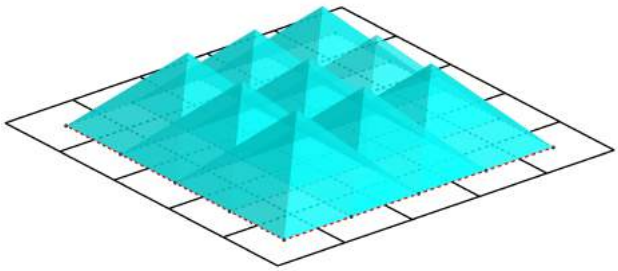
$$P_f = \underset{\sim}{P}_{op} P_c \tag{4.4}$$

where P_f is a $n_f \times 1$ fine-scale pressure solution, P_c is a $n_c \times 1$ coarse-scale pressure solution and $\underset{\sim}{P}_{op}$ $n_f \times n_c$ matrix.

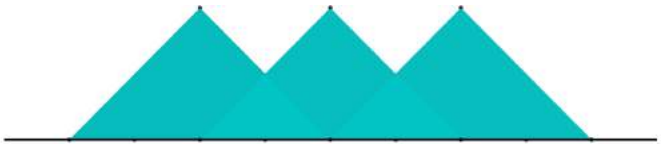
Figure 15 – Basis Function Sketching



(a) Primal and Dual Coarse Grid generating the Support Regions



(b) Isometric Representation of General Basis Function on a two-dimensional domain



(c) Basis Function Overlaping on a Cross Section: Basis function ranging from 1 to 0

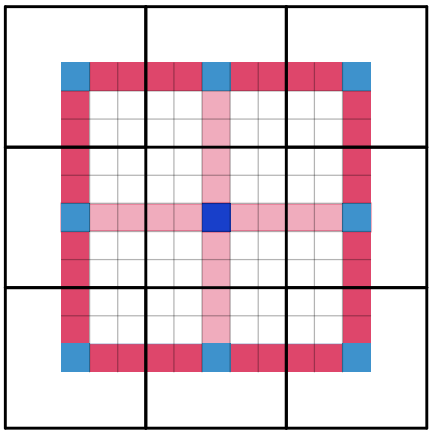
Source: Author.

The basis function are computed by solving the homogeneous elliptic part of Equation (2.9) using a set of Reduced Boundary Conditions devised to decouple the domain generating multiple smaller problems. As a result, the basis functions associated to a coarse cell center can be interpreted as normalized pressures with values varying from 1 on the coarse cell center to 0 outside the zone of influence, or the support of this cell. Thus, the j^{th} column in $\underset{\sim}{P}_{op}$ stores how the coarse cell center (x_j^P) influence the high-resolution domain. This column becomes a $n_f \times 1$ vector with non-zero values only inside the support region of the associated basis function. Additionally, the i^{th} row is comprised of all the weights to the $1 \times n_c$ coarse pressure solution vector used to span the pressure solution on the Ω_i^f fine-scale volume.

Once more, a consistent Finite Volume approximation requires that flux must be conserved. Regarding the Prolongation Operator, this is done in two different ways. The basis functions must be calculated using a conservative flux approximation and the basis functions must ensure partition of unity, that is, the sum of the values of all basis function in any point inside the domain is always equal to 1. In other words, no information originated on the coarse-scale solution is lost as they are projected on the fine-scale. The basis functions are weights of a convex combination of the coarse-scale space. Therefore, the sum of any i^{th} row is equals to 1.

Figure 15 illustrates the behavior of the basis functions. On Figure 15a we present a coarse grid (black lines) along with its dual (red dashed lines). When combining the four dual volumes surrounding any coarse center we have the support of a base function associated with each coarse volume. As we previously mentioned, the support regions

Figure 16 – Support Region



Source: Author.

overlap, therefore the basis function also overlap. Figure 15b shows the basis function associated with the inner volumes of the primal mesh. Notice how each basis function expand only inside small portions of the domain peaking at each coarse center. This zone of influence crosses the boundaries of each coarse volume reaching neighbor coarse volumes. The cross section view on Figure 15c shows that basis function idea is intrinsically related to the Finite Element concepts of hat functions.

Regardless of the choice for the basis function, the Prolongation Operator is given by:

$$(\underline{P}_{op})_{i,j} = \phi^j(x_i) \quad \text{where } 1 \leqslant j \leqslant n_c \quad \text{and } 1 \leqslant i \leqslant n_f \tag{4.5}$$

The basis functions used by Jenny, Lee and Tchelepi (2003) on the original MsFVM are very similar to those in the MsFEM (HOU; WU, 1997). However, they are calculated locally on each dual coarse volume. By definition, the resulting velocity field is mass conservative on the coarse-scale but not inside the fine-scale velocity field. This happens as a consequence of the boundary conditions used to decouple the problem. As the normal flow outside the support region is neglected, the velocity field across the dual block interfaces becomes discontinuous. To deal with this issue, Jenny, Lee and Tchelepi (2003) proposed solving another set of basis function to reimpose conservation. A conservative fine-scale velocity field is obtained by comprising the already conservative velocity field calculated on the boundaries of each coarse volume and the velocity field found inside the coarse volumes by using this conservative field as Neumann Boundary conditions on each coarse cell. In this approach, the basis function associated with a coarse cell j is defined as:

$$\begin{cases} -\vec{\nabla}\cdot(-\lambda\underset{\sim}{K}\vec{\nabla}\phi_i^j) = 0 \quad \text{in} \quad \Omega_{A_j}^d \\ -\frac{\partial}{\partial x_t}\left(-\lambda\underset{\sim}{K}\frac{\partial}{\partial x_t}\phi_i^j\right) = 0 \quad \text{on} \quad \partial\Omega_{A_j}^d \\ \phi_i^j = \delta_{ix_j^P} \quad \text{on} \quad V(\partial\Omega_{A_j}^d) \end{cases} \tag{4.6}$$

where t stands for the component tangential to the boundary $\phi_i^j = \phi^j(x^i)$, A_j stands for all the dual cells around x_j^P and $V(\partial\Omega_{A_j}^d)$ stands for the vertices in these duals.

The boundary conditions used to solve the problem given by equation (4.6) are called the Reduced Boundary Conditions. Figure 16 presents a sketch of a support of a coarse cell center A_j (dark blue) to help better understand these conditions. Briefly they mean that for each coarse volume j (bold solid lines), a one-dimensional problem (light pink region) is solved setting 1 on x_j^P (dark blue) and 0 on the surrounding centers(light blue), to be used as boundary conditions to solve the rest of the support region (white cells).

Each method has a particular choice for basis functions and Reduced Boundary Conditions. Traditional MsFV methods generally employ TPFA to solve Equation (4.6) (JENNY; LEE; TCHELEPI, 2006; ZHOU, 2010) that have proven to be accurate, efficient and robust for multiphase-flow simulation on highly heterogeneous and isotropic or moderate k-orthogonal reservoirs. The idea of Multiscale Methods is to exchange accuracy, to the detriment of computational cost. Theoretically, Multiscale Methods could converge to the numerical solution of any PDE, as long as, the approximation of Equation (4.4) is exact (ZHOU, 2010).

4.1.3 Algebraic Formulation of the MsFVM

As we have previously mentioned, multiscale methods project the fine-scale system of equations onto a coarse-scale space to be solved and projected back. For that, we must apply a Finite Volume Flux Approximation to discretize the flux on Equation (3.5) obtaining a fully discrete system of equations:

$$\underset{\sim}{T}_f P_f = Q_f \quad \text{for} \quad P_f \in \Omega^f \tag{4.7}$$

where P_f is a $n_f \times 1$ vector containing the pressure for all n_f volumes in Ω^f, Q_f a vector containing discrete representation of sink and source term in Ω^f and $\underset{\sim}{T}^f$ is the $n_f \times n_f$ transmissibility matrix.

From this point, we use the definition of Prolongation Operator presented on Equation (4.4) in Equation (4.7) and pre multiply both sides of the equation by the

Restriction Operator, resulting in:

$$\underset{\sim}{R}_{op}\underline{T}_f\underline{P}_{op}P_c = \underset{\sim}{R}_{op}Q_f \tag{4.8}$$

We use the definition of Restriction Operator on the LHS of the equation (4.2) to define source/sink term vector on the coarse-scale:

$$\underset{\sim}{R}_{op}\underline{T}_f\underline{P}_{op}P_c = Q_c \tag{4.9}$$

which is a $n_c \times n_c$ coarse-scale system of equations:

$$\underset{\sim}{T}_c P_c = Q_c \tag{4.10}$$

where

$$\underset{\sim}{T}_c = \underset{\sim}{R}_{op}\underset{\sim}{T}_f\underset{\sim}{P}_{op} \tag{4.11}$$

$$Q_c = \underset{\sim}{R}_{op}Q_f \tag{4.12}$$

By solving Equation (A.16), we have:

$$P_c = \underset{\sim}{T}_c^{-1}\underset{\sim}{Q}_c \tag{4.13}$$

Thus, we construct the multiscale solution by projecting it back onto the fine-scale space:

$$P_{ms} = \underline{P}_{op}P_c = \underline{P}_{op}\underset{\sim}{T}_c^{-1}\underset{\sim}{Q}_c \equiv \underline{P}_{op}(\underline{R}_{op}\underline{T}_f\underline{P}_{op})^{-1}\underline{R}_{op}Q_f \tag{4.14}$$

This algebraic form of interpreting Multiscale Methods was first proposed by Zhou and Tchelepi (2008) to expand the multiscale operator adding more complex physics i.e: compressibility, gravity and capillarity. As the multiscale solution is obtained by using only matrix operators, this scheme is referred as the operator based multiscale method (OBMM). Note that the procedures used to derive Equation (4.14) are mesh independent, in contrast to classical MsFVM (JENNY; LEE; TCHELEPI, 2003; JENNY; LEE; TCHELEPI, 2006) that use underlying grid information to construct a dual mesh used to derive the coarse-scale transmissibility field. This means that in order to extend MsFVM to work on unstructured grids it is only necessary do devise multiscale operators accordingly.

4.1.4 Velocity Field Calculation

The difficulty in finding a consistent approximation for the transport equation lays on the fact that it is mandatory to obtain a conservative velocity field in all grids.

Finite Volume Approximations are by definition conservative, however, the multiscale pressure solution derived in the last subsection gives birth to a velocity field that is only locally conservative, which means the flow is discontinuous on the boundaries of the dual coarse volumes but continuous inside. This is due to the use of the set of boundary conditions, to solve Equation (4.6), that neglected normal flux on the surface of the dual cells in order to uncouple the domain. Therefore, methods consistent with the Multiscale Finite Volume family need to use some sort of reconstruction step in order to obtain a fully conservative fine-scale velocity field. Figure 17 illustrates the flux reconstruction algorithm proposed by Jenny, Lee and Tchelepi (2006) and used in this thesis. This is done in two straightforwards steps. As a consequence of fluxes being discontinuous on surfaces of the support region, the interface coarse volumes remains conservative. The first step is to use the pressure field obtain by Equation 4.14 to retrieve on the surface of each primal coarse volume, in order to use them as Neumann Boundary Conditions to solve Equation (2.6) restricted to each primal coarse volume. The second step is to use the resulting pressure fields to calculate a new velocity field in all surfaces but on the ones contained on the contour of each primal coarse volume as described in Equation4.15. The reconstructed flux is a composition of the velocities found in these two steps. For the boundaries of each coarse volume, we use the velocity calculated on the first step, inside, we use the velocity computed in the second step.

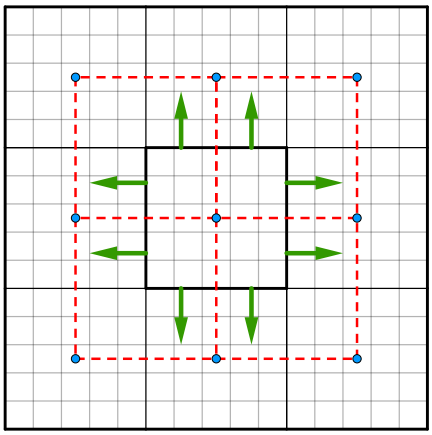
$$\begin{cases} -\vec{\nabla}\cdot(-\lambda\underline{K}\vec{\nabla}(p_{msc}) = q_f & \text{inside} \quad \Omega^p \\ -\frac{\partial}{\partial x_n}\left(-\lambda\underline{K}\frac{\partial}{\partial x_n}(p_{msc})_i^j\right) = \vec{v}_{ms}\cdot\vec{N} & \text{on} \quad \partial\Omega^p \end{cases} \tag{4.15}$$

where n is the normal component of the flow on the boundary of the primal coarse volumes, p_{msc} is the new pressure field calculated inside each primal coarse volume and $\vec{v}_{ms}\cdot\vec{N}$ the Neumann flux computed on the surface of each primal coarse volume using the pressure field calculated using Equation 4.14.

4.1.5 Multiscale Errors

By construction, multiscale methods rely on the numerical scheme used to discretize the flux on the underlying high-resolution mesh. Therefore, the quality of multiscale solution is always bounded by direct simulation on the fine grid or reference solution, which becomes a theoretical upper limit. According to Equation (3.56), the variation of the velocities between solutions in two-phase flow simulations using an IMPES approach may result in two simulations with different numbers of time step. In other words, comparing results of the multiscale and reference solution after the same number of time step means comparing results calculated in different times. Thus, for a proper comparison the pressure and saturation field are evaluated after the same amount of simulation time.

Figure 17 – Neumann



Source: Author.

In this thesis, we employ the same metric used by Barbosa (2017), Moyner and Lie (2015), Jenny, Lee and Tchelepi (2003), Jenny, Lee and Tchelepi (2006), Zhou (2010), Hajibeygi and Jenny (2011) to assess the accuracy of the MsCV framework. Let us now determine the errors used in this thesis. First, we define respectively the L_2 and L_∞ norms as:

$$||\vec{x}||_2 = ||\vec{x}|| = \sum_{i=1}^{N} |x_i|^{\frac{1}{2}} \tag{4.16}$$

$$||\vec{x}||_\infty = \max_{1 \leq i \leq N} (|x_i|) \tag{4.17}$$

where x_i is the i^{th} component of a N size vector $\vec{x}$.

The absolute and relative errors are defined as:

$$e_{abs} = ||X^{ref} - X^{ms}|| \tag{4.18}$$

$$e_{rel} = \frac{||X^{ref} - X^{ms}||}{||X^{ref}||} \tag{4.19}$$

where X^{ms} and X^{ref} are the multiscale and reference solution for a generic variable X, respectively.

4.2 General Multiscale Algorithm for Two-Phase Flow

In the previous sections we have introduced basic concepts used to derive general Multiscale Finite Volume Methods. Now, we summarize these ideas by presenting a

general algorithm to simulate two-phase flows using a IMPES strategy which replaces the implicit pressure solver with a MsFVM scheme.

Algorithm 1 Multiscale Algorithm for the Pressure Equation in Two-Phase Flows

procedure PREPROCESSING ALGORITHM(Input: Ω Physical Domain)
  $\Omega^f \leftarrow$ FINE MESH DISCRETIZATION ALGORITHM(Ω^f)
  $\Omega_p^c \leftarrow$ COARSENING ALGORITHM(Ω^f)
  $x^P \leftarrow$ CALCULATE PRIMAL COARSE CELL CENTERS(Ω_p^c)
  $\Omega_d^c \leftarrow$ DUAL MESH GENERATION($\Omega^f, \Omega_p^c, x^P$)
  $(S, K, Q_{wells}, t_{max}, BC) \leftarrow$ READ INITIAL SIMULATION PARAMETERS ▷ Initial Saturation Field, Permeability Field, Wells, Maximum Simulation Time, Boundary Conditions
end procedure
procedure START SIMULATION(Input: Ω^f,Ω_p^c,x^P, $\Omega_d^c, S, K, Q_{wells}, t_{max}, BC$)
  $\underline{R}_{op} \leftarrow$ GENERATE RESTRICTION OPERATOR(Ω^f,Ω_p^c)
  while $t \leqslant t_{max}$ **do**
    $(\lambda_f, \lambda_c) \leftarrow$ CALCULATE MOBILITY(S, K, Q_{wells}) ▷ Fine/Coarse Scale Fields
    procedure PRESSURE EQUATION(Input: Ω^f,Ω_p^c,x^P, $\Omega_d^c, S_o, K, Q_{wells}, \lambda_f, \lambda_c, BC, R_{op}$)
      $(T_f, Q_f) \leftarrow$ FINE-SCALE DISCRETE SYSTEM ASSEMBLY ($\Omega^f, S, Q_{wells}\lambda_f, BC$)
      $\underline{P}_{op} \leftarrow$ GENERATE PROLONGATION OPERATOR(T_f)
      $P_c \leftarrow (\underline{R}_{op}\underline{T}_f\underline{P}_{op})^{-1}\underline{R}_{op}Q_f$ ▷ Calculate Coarse-Scale Pressure Field
      $P_{ms} \leftarrow \underline{P}_{op}P_c$ ▷ Calculate Fine-Scale Pressure Field
      $V_c^{cons} \leftarrow$ CALCULATE VELOCITY(P_{ms}, K, λ_f) ▷ Calculate Fine-Scale Velocity Field only conservative on Primal Coarse Cells
      procedure FLUX RECONSTRUCTION(Input: $T_f, Q_f, P_c, V_c^{cons}, K, \lambda_f, \lambda_c$)
        for each volume k in Ω_p^c **do**
          $(T_c^k, Q_c^k) \leftarrow$ COASE-SCALE SYSTEM ASSEMBLY ($T_f, Q_f, P_c, V_c^{cons}, \lambda_c$)
          $p_{neumann} \leftarrow (T_c^k)^{-1}Q_c^k$ ▷ $p_{neumann}$ stores pressure field
        end for
        $V_p^{cons} \leftarrow$ CALCULATE VELOCITY($P_{ms}, K, \lambda_f, \lambda_c$) ▷ Calculating Fine-Scale Velocity Field only conservative inside Primal Coarse Cells
        $V_{ms} \leftarrow$ ASSEMBLY MULTISCALE VELOCITY FIELD(V_p^{cons}, V_c^{cons}) ▷ Fully Conservative Fine-Scale Velocity Field
      end procedure
    end procedure
    $f_{flux} \leftarrow$CALCULATE FRACTIONAL FLUX (S, V_{ms})
    $\Delta t \leftarrow$ MAXIMUM TIME STEP(f_{flux}, V_{ms}, S) ▷ Calculate Time Step
    procedure SATURATION EQUATION(Input: f_{flux}, V_{ms}, S, BC)
      $S \leftarrow$ SATURATION UPDATE(f_{flux}, V_{ms}, S, BC)
    end procedure
    $t \leftarrow t + \Delta t$
  end while
end procedure

4.3 Multiscale Restriction Smoothed-Basis method coupled with a MPFA-D solver

In this subsection, we use the formulation presented on the previous subsection to present the Multiscale Restriction Smoothed-Basis Method. As most of the basic concepts of the MsRSB share the basic principles with the MsFV, we focus on presenting the differences by comparing these methods and introducing the algorithms used to handle general unstructured meshes. Algorithm 1 is used as guide to describe the MsRSB along with the technical details of its implementation.

First, we start this subsection introducing basic concepts of the MsRSB method and comparing with the MsFV. We continue by presenting the multiscale preprocessing stage algorithms: coarsening algorithms, coarse cell center calculation, support region generation. Finally, we introduce the MsRSB method, how the multiscale operators are calculated and we comment on the coupling with the MPFA-D solver. We call the resulting framework the Multiscale Control Volumed, a scheme capable of handling unstructured grids on both scales.

4.3.1 Basic MsRSB Concepts

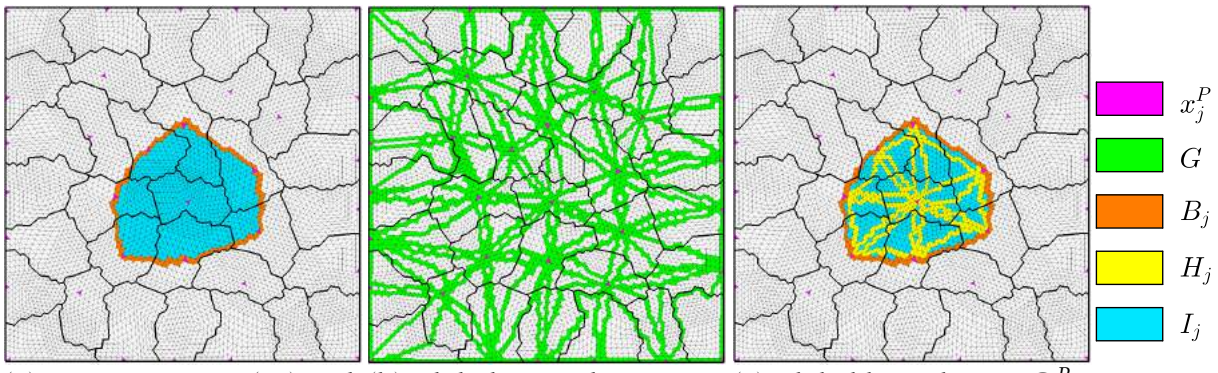
MsRSB inherits most of the geometric entities defined for the MsFVM. Concepts such as fine scale mesh , primal coarse mesh, primal coarse center and the support of a coarse center remain identical, however the means to obtain them may vary. More on that on subsection (4.3.2). One of the major difference between these methods concerns the support region. While the MsFVM uses it implicitly, the MsRSB defines it explicitly and uses the very definition to calculate the prolongation operator direct on it. Hereby, we present the basic geometric concepts used by the MsRSB.

Support Region of a Primal Coarse Volume j (I_j): The definition of the support region of a primal coarse volume is identical to the MsFVM. It consists in all fine-scale volumes where:

$$(\underset{\sim}{P}_{op})_{i,j} > 0 \quad \forall i \in I_j, \quad \text{otherwise} \quad (\underset{\sim}{P}_{op})_{i,j} = 0 \tag{4.20}$$

I_j defines a influence zone of the coarse cell center x_j^P. Unlike the MsFVM that uses the dual coarse volumes to generate the support region, the MsRSB uses the primal coarse cells and the coarse center to create the support region of a primal coarse volume in order to define all other geometric entities. Figure 18a illustrates the support region of the center coarse cell as the blue volumes around it. Note that the pink fine control volume, center of the middle coarse volume, is not included on the support region.

Figure 18 – Basic concepts of a general coarse volume Ω_j^P MsRSB



(a) Support region (I_j) and (B_j) support boundary.

(b) Global Boundary, $G = B_1 \cup B_2 \cup B_3 \cup ... \cup B_k$.

(c) Global boundary in Ω_j^P $H_j = I_j \cap G$.

Source: Author.

Support Boundary Region, (B_j)**:** The support of a boundary region consists in all volumes that share at least one edge with I_j but are not contained in it. Figure 18a and 18c depict these as orange volumes. It is worth noting that no primal coarse center x_j^P (in pink) are included in this group.

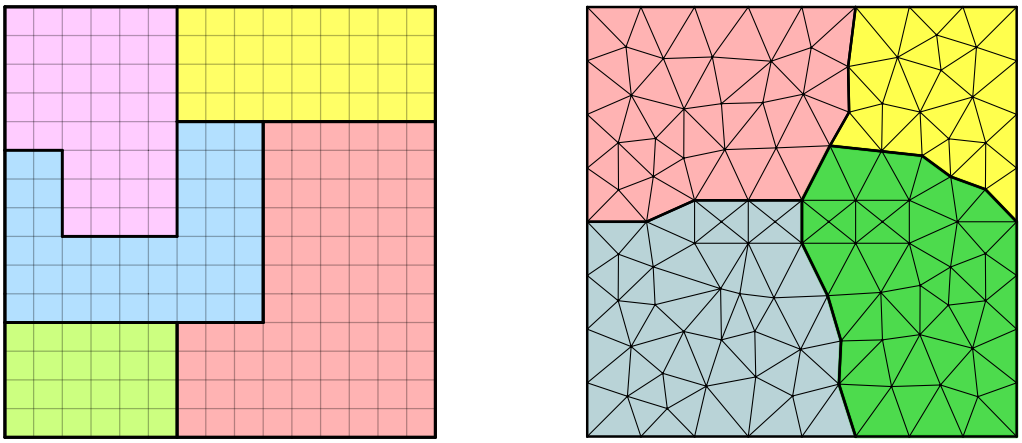
Global Support Boundary, (G)**:** The Global Support Boundary Region is the union of all support boundaries associated with all coarse volumes of the primal coarse mesh. Figure 18b illustrates as the green fine-scale volumes.

Global Support Boundary in a Support Region, H_j**:** The region consists in the intersection of a general I_j with G. They are represented as the yellow volumes on Figure 18c.

4.3.2 Preprocessing Algorithms

In order to make a MsCV compatible with general grids, we must understand the peculiarities these grids possess to propose algorithms to generalize the ideas behind the MsFVM. The first main difference is the fact the MsRSB allows coarse volumes to assume general non-regular shapes. This makes the coarsening algorithm not trivial as agglomerating fine-scale cells in rectangular volumes. Also, the extension we propose to overcome the limitation of the classical MsRSB that used TPFA as the pressure solver, reducing the choice of underlying fine-scale grids. Therefore, the second main difference is, fine-scale volumes may also be unstructured. Figure 19 describes these differences. Note that in Figure 19a, the coarse volumes are unstructured but the fine-scale volumes are not. Figure 19b describes a general grid with unstructured fine-scale and coarse-scale volumes.

Figure 19 – General unstructured grids.



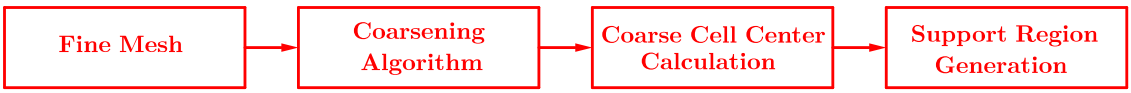
(a) Structured fine mesh with unstructured primal coarse grid.

(b) Unstructured fine mesh with unstructured primal coarse grid.

Source: Author.

As a consequence, we must define coarsening algorithms, calculate the primal cell centers to define the support region. The preprocessing stage consists in all these algorithms. See Figure 20. For the fine Mesh discretization step, we use Gmsh (GEUZAINE; REMACLE, 2009), a finite-element mesh generator broadly used in the numerical simulation context.

Figure 20 – Preprocessing Algorithm

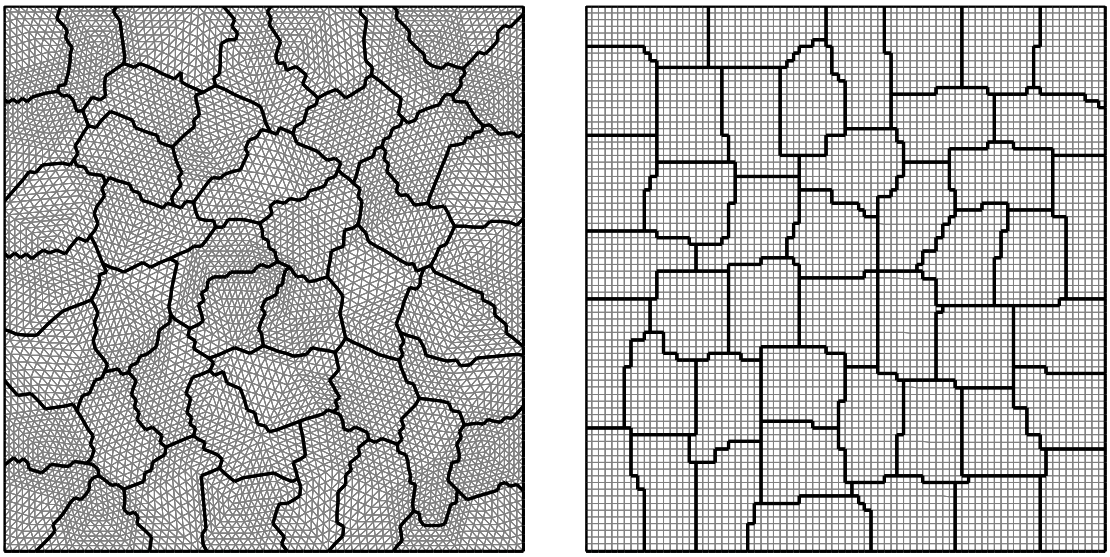


Source: Author.

4.3.2.1 Coarsening Algorithms

In this thesis, we use two different approaches for generating a primal coarse-scale mesh. The first approach used Metis (KARYPIS; KUMAR, 2009), a set of partitioner tools used for distributing workload among different cores in parallel computing simulations, this gives birth to coarse grids that assume unstructured shapes. Figure 21 shows two examples of meshes generated using Metis algorithm. On the other hand, since Metis was not conceived for multiscale coarsening their grids may cause oscillations as explained on the Results. In addition, Metis may generate coarse grids in which a fine-scale element from a random partition is completely surrounded by fine-scale elements of other partitions (See Figure 22). In other words, it does not ensure that all cells in the same partition share at least one face.

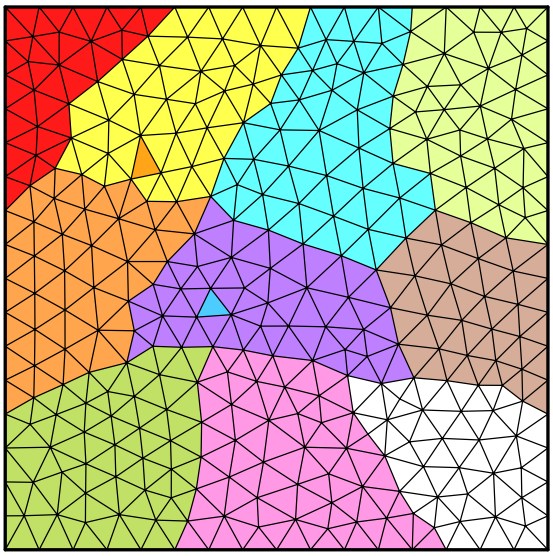
Figure 21 – Coarse Meshes with 45 coarse volumes generated using Metis.



(a) Underlying unstructured fine-scale grid. (b) Underlying structured fine-scale grid.

Source: Author.

Figure 22 – Coarse mesh generated with Metis improper for multiscale simulation.

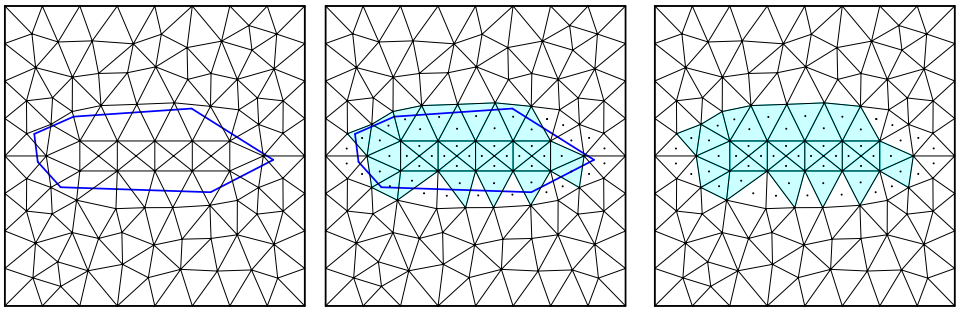


Source: Author.

The need for another simple yet robust coarsening algorithm, made us develop some partitioning routines that allows users to literally draw the contours of the coarse volumes. This geometrical approach showed better results in most cases in comparison to coarse girds generated using Metis as we are capable of generating coarse grids that better model the geometry of the problem. Check the Results of this thesis for more on that subject.

The core idea to these routines is an algorithm that checks whether the centroid of a fine-scale cell is inside of a general polygon. For the sake of simplicity name this algorithm *inpolygon*. Figure 23 describes this process. It works by taking as an input a generally defined polygon and multiple fine-scale cells (Figure23a). This routine checks the centroid of each of these cell and returns the cells in which their centroids lye inside this polygon (Figure23b e 23c).

Figure 23 – Inpolygon Algorithm: Checks if a center is inside a polygon.



(a) Input: General polygons and fine-scale cells.

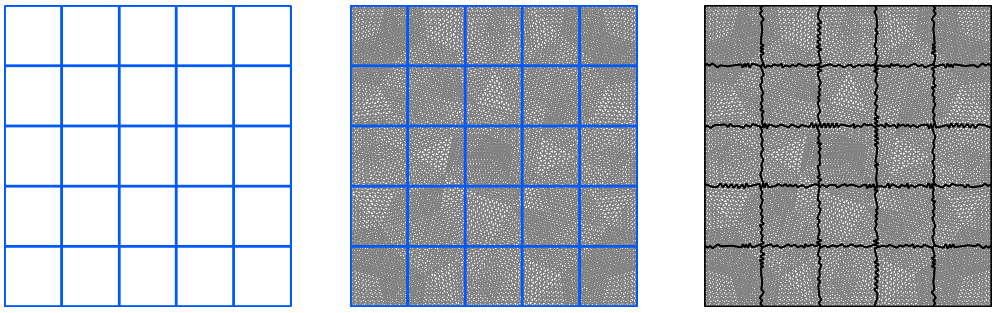
(b) Checks which fine-cell center are inside the polygont.

(c) Return: Fine-scale cells inside this polygon.

Source: Author.

The first algorithm developed creates a base coarse mesh using simple geometric shapes. This is done by drawing upon the fine-scale mesh these forms, then, inpolygon is used to check if each fine cell belongs to each coarse cell. Figure 24 uses the underlying fine-scale mesh on Figure 21a to illustrate this process. First, the user inputs which basic shape will be used along with the number of elements in the X and Y axes, this creates a basic forming grid (Figure 24a) that is applied upon a fine-scale mesh (Figure 24b). Each volume in this basic forming coarse grid uses inpolygon to check the fine-scale volumes that belong to them. As a consequence, the resulting coarse-scale have contours that resembles but are not necessarily equal to those in the original forming grid (Figure 24c). Figure 25 repeats the same process using an hexagonal basic grid.

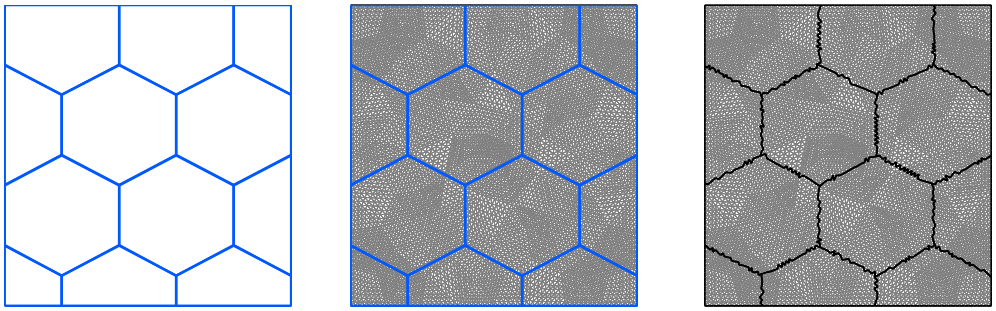
Figure 24 – Coarse mesh generation.



(a) User defines a basic forming grid. (b) Apply forming grid to a fine-scale mesh. (c) Generate coarse-scale volumes.

Source: Author.

Figure 25 – Hexagonal coarse mesh generation.

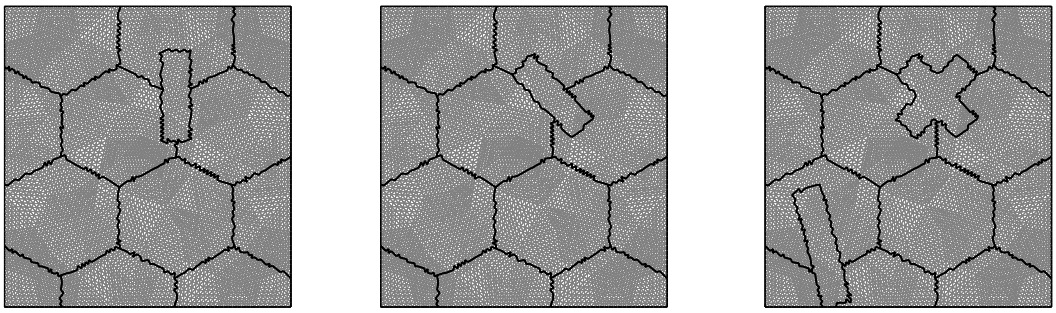


(a) User defines a basic forming grid. (b) Apply forming grid to a fine-scale mesh. (c) Generate coarse-scale volumes.

Source: Author.

We have also implemented a routine that let the user to draw different shapes on the coarse-scale grid, which in turn uses inpolygon to create additional coarse volumes. This allows the user to generate complex and non-conforming coarse-scale meshes taking into account the physical and geometrical properties of the media. Figure 26 describes some of its features. It possible to input the size, position where a basic shapes is applied (Figure 26a). User can rotate and manipulate (Figure 26b) as well as create as many shapes as needed to compose a complex cell (Figure 26c).

Figure 26 – Support and Boundary Support Region generation for general unstructured grids.



(a) Position shapes. (b) Rotate shapes. (c) Create complex volumes.

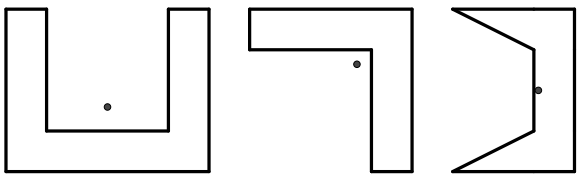
Source: Author.

Finally, after using these steps we have developed an algorithm that checks the coarse-scale mesh integrity. In other words, a routine that ensures that all fine-scale volumes in a coarse-volume shares at least one edge with each other. The fine-scale volumes that do not bear these conditions are reallocated to neighbor coarse-volumes.

4.3.2.1.1 Coarse Cell Center Calculation

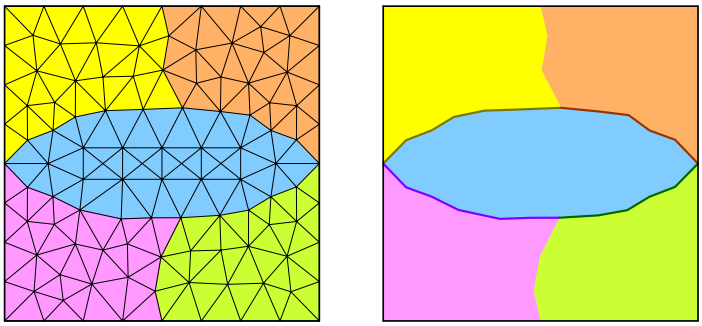
The process of defining the primal coarse cell center in most of classical MsFVMs is as simple as calculating the centroid. However, for unstructured coarse cells this is not an option as the centroid of non-convex shapes may lay too close to the contours of the polygon or even outside it. (See figure 27.) We use an unstructured coarse mesh with a unitary domain in Figure 28 to illustrate a 3 step algorithm that deals with this limitation. Let us calculate the center of the blue coarse mesh in Figure 28a. The first step is to find the interfaces between the blue coarse volume and its neighbors (See Figure 28b).

Figure 27 – Centroid of non-convex coarse cells.



Source: Author.

Figure 28 – Blue coarse volume interfaces.

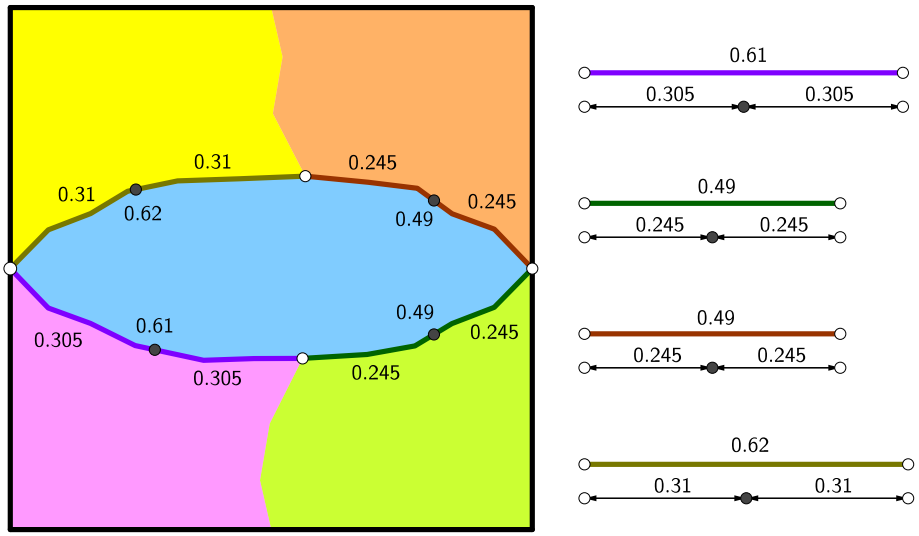


(a) Ordinary coarse mesh. (b) Highlighted interfaces.

Source: Author.

Second, one must find the middle of these interface. This is done by finding the point that splits in two the length of each interface Figure depicts 29.

Figure 29 – Finding the center of the interfaces.



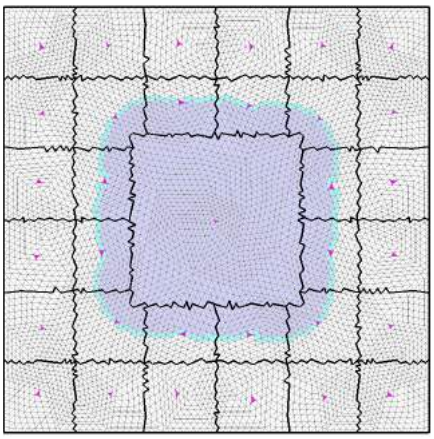
Source: Author.

Finally, we use these points to calculate the Geometric Median. By definition, this median finds a point which minimizes the sum of the distances to the sample points. That is to say, the center point of a coarse volume is the point which has the lowest sum of distances to the middle edges interfaces. This ensures that the center always lays inside the coarse volume. In this thesis the geometric median is calculated using classic Weiszfeld's algorithm (WEISZFELD, 1937). The primal coarse center (x^P) is the fine-scale cell whose center lays closest to the geometric median.

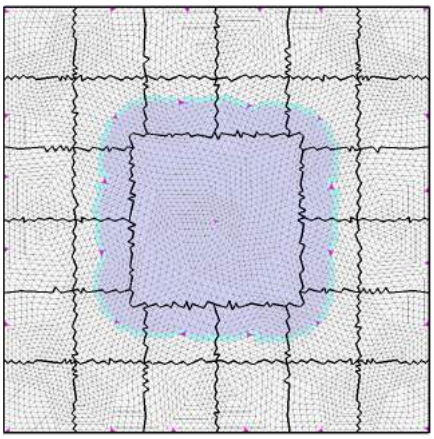
Barbosa (2017) suggests relocating the center of primal coarse cells lying on the boundary of the domain to the fine-scale volume closest to the intersection between the

coarse cell surface and the physical domain. Barbosa conduced a series of simulations showing that this modification increases the accuracy of the resulting pressure field. Thereby, we have also implemented these modifications.

Figure 30 – Relocated primal coarse cell center in a non-conforming unstructured mesh.



(a) Classic primal coarse cell center of coarse cells lying on the boundary.



(b) Relocated primal coarse cell center of coarse cells lying on the boundary.

Source: Author.

4.3.2.2 Support Region Generation

The final step of the preprocessing stage of most MsFVM is to generate an auxiliary mesh to be used to reconstruct a conservative flux field throughout the fine mesh. While the MsFVM uses the dual mesh, the MsRSB uses the support region of each coarse cell I_j for this purposes. As we work with unstructured meshes, there are several ways to define this auxiliary mesh, each with advantages and disadvantages that directly impact the quality of the solution.

In this thesis, we use the algorithm proposed by Moyner and Lie (2015) with a slight modification to better work with fine-scale unstructured grids to generate the support region of each primal coarse volume. Figure 31 illustrates the process that generates the support region and the support boundary associated with the center primal coarse volume whose coarse volume is green.

(a) The first step is to define which primal coarse volumes are neighbors of the center coarse volume. We define neighbor as volumes that share at least one node.

(b) We find the middle of the interface of the central volume with each of its neighbors (blue nodes). We also find the middle of the interface among all neighbor volumes (red nodes).

(c) We use these points plus the center points of the neighbor coarse cell center (black nodes in pink volumes) to perform a Delaunay triangulation.

(d) For a consistent support region, no neighbor coarse cell center may lay inside of this region. Then, we perform a search among the triangles generated by the Delaunay triangulation to find those that do not meet this condition. In other words we find those triangles that prevent the coarse cell center nodes (black nodes) to be on the boundary of the resulting shape.

(e) We remove the triangles found on the previous step from the Convex Hull of the points found at the (b) step.

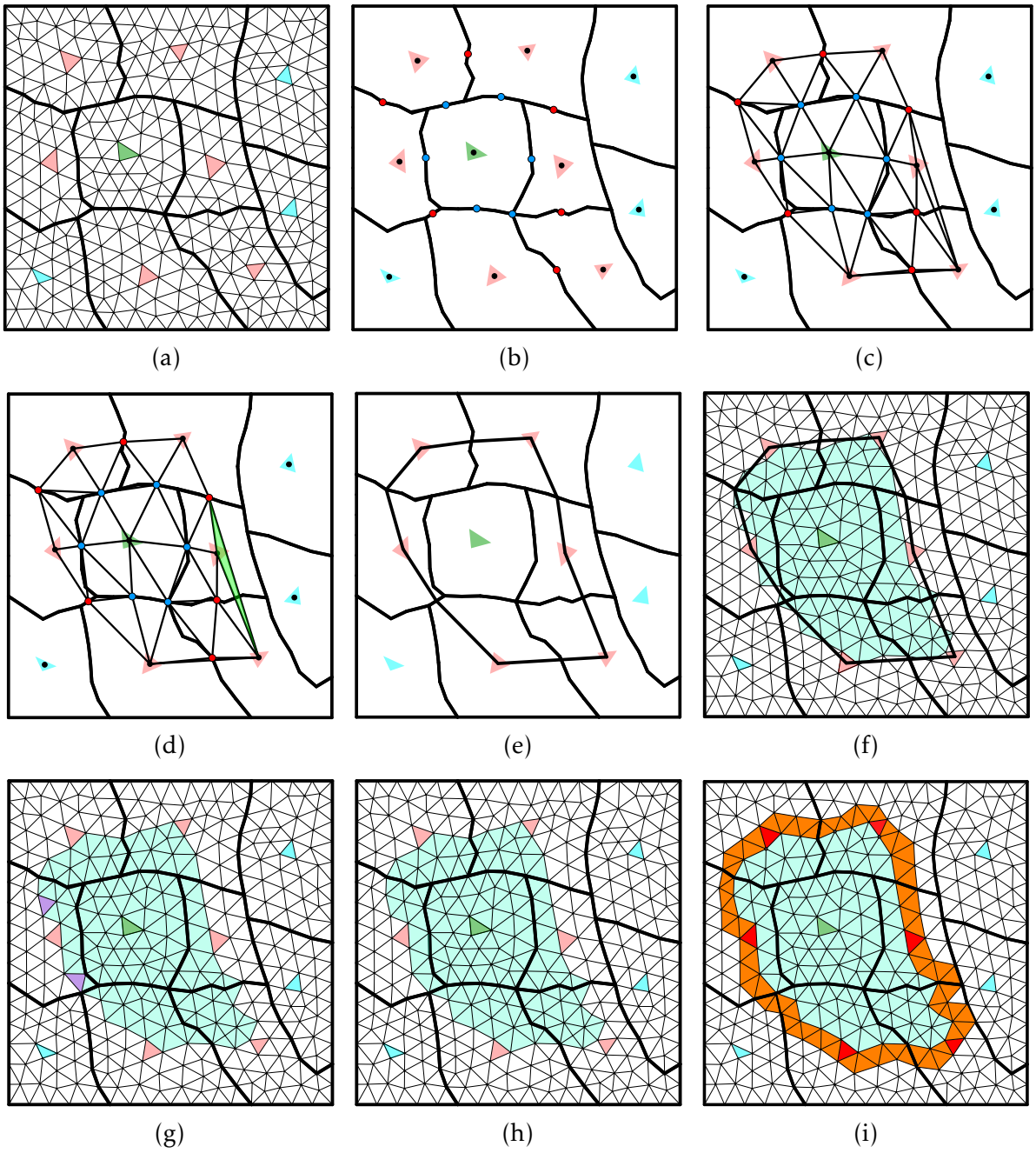
(f) We use the basis algorithm using the resulting polygon to find which fine-scale volumes lay inside it (blue volumes).

(g) We search for fine-scale volumes that share at least two edges with this region.

(h) We add these volumes to the blue region to formally define the support region of the middle coarse volume.

(i) Finally, we define as the support boundary all volumes that share at least one node with the volumes inside the support region.

Figure 31 – Support region and boundary support region generation for general unstructured grids.



Source: Author.

This process is repeated for each coarse volume on the primal coarse mesh. After this, we use the very definition to find the Global Support Boundary and Global Support Boundary in a Support Regions (See Section 4.3.1).

4.3.3 MsRSB Multiscale Operators

Besides the modifications in the preprocessing stage, the major difference between the MsRSB and the MsFVM lays on the way the prolongation operator is generated. While the MsFVM finds it implicitly by using the classic Reduced Boundary Conditions to solve Equation (4.6), the MsRSB calculates it explicitly using a modified version of Reduced Boundary Conditions to enable simulations on unstructured coarse-scale meshes. Moreover, the basis functions of the MsRSB are computed on the support regions which are significantly bigger than the four dual where the basis functions are solved on the classic MsFVM (See Figure 16). Cost wise, it is expensive to solve an implicit simulation at every time step. The MsFV deals with it using adaptive computation conditions to determine when to update the prolongation operator (JENNY; LEE; TCHELEPI, 2006). Two-phase simulation using the MsRSB as pressure solver benefit greatly as this method uses explicit prolongation operator. No more than a 100 iterations are needed to recalculate the prolongation operator found on the previous time step decreasing significantly the coast associated with updating the basis functions. Regarding the Restriction operator,the MsRSB uses the same as the MsFV which is defined in Equation (4.3).

4.3.3.1 Prolongation Operator

The MsRSB calculates the prolongation operator in a very forward manner. Unlike the classic MsFVM that assembles the fine-scale transmissibility discrete system and multiple basis function problems to calculate a multiscale solution, the MsRSB uses the fine-scale system to calculate the basis function. This is done a pre-conditioner on the fine-scale matrix which strips its original boundary conditions. The preconditioned matrix is then used in a modified Weighted Jacbobi method which employs a different set of boundary conditions to calculate the basis functions. See Figure 32.

Figure 32 – Preprocessing Algorithm.

Source: Author.

Therefore, adapting the MsRSB to work with different FV formulations become as simple as assemble a fine-scale system with an accordingly flux approximation and finding a conservative reconstructed velocity field. In this thesis, we employed the flux approximation derived in the Section 3.2.2.2 to assemble the MPFA-D.

Let $\underset{\sim}{T}_f$ be a $n \times n$ transmissibility matrix of Equation (4.7). We define the preconditioned matrix $\underset{\sim}{T}_f^{pre}$ as:

$$(\underset{\sim}{T}_f^{pre})_{ij} = \begin{cases} (\underset{\sim}{T}_f)_{ij} & i \neq j; \\ (\underset{\sim}{T}_f)_{ij} - \sum_{k=1}^{n,k\neq j} (\underset{\sim}{T}_f)_{ik} & i = j; \end{cases} \tag{4.21}$$

As we have previously mentioned, the MsRSB basis functions are solved iteratively inside each support region. As iterative methods require an initial solution, we must choose an initial guess to ensure that partition of unity is not lost while iterating. One natural option is to use the indicator function of each corresponding coarse volume. In other words, one is set to all fine-scale volumes inside the associated coarse volume, i.e. :

$$(\underset{\sim}{P}_{op}^0)_{ij} = \begin{cases} 1 & \Omega_i^f \in \Omega_j^c; \\ 0, & \text{otherwise} \end{cases} \tag{4.22}$$

Note that Equation (4.22) is equivalent to:

$$(\underset{\sim}{P}_{op}^0)_{ij} = \underset{\sim}{R}_{op}^T \tag{4.23}$$

For IMPES like simulations where the MsRSB is used exclusively as a solver for the pressure equation, the prolongation operator computed on the previous time-step is used as an initial guess instead of Equation (4.22). For such problems, the mobilities are updated slowly resulting in little changes to the fine-scale transmissibility matrix. As a consequence, the prolongation operator tends to converge very quickly after the first time-step.

The Prolongation Operator is finally defined using the following procedure (MOYNER; LIE, 2015):

1. We calculate the classical Weighted Jacobi increment.

$$\widehat{d_j} = -\omega D^{-1} \underset{\sim}{T}_f^{pre} (\underset{\sim}{P}_{op})_j^{\eta} \tag{4.24}$$

where is D is the diagonal matrix containing the main diagonal of the preconditioned transmissibility matrix $\underset{\sim}{T}_f^{pre}$ and ω is the damping parameter, which is set to 2/3 the optimal choice for solving Poisson's equation with constant coefficients using the Jacobi's method Moyner and Lie (2015).

2. The following piecewise function modifies the increment ensuring that partition of unity is not lost and restricting growth to the inner side of the support region.

$$d_{ij} = \begin{cases} \frac{\widehat{d}_{ij} - (\underline{P}_{op})^{\eta}_{ij} \sum_{k \in H_k} \widehat{d}_{ik}}{1 + \sum_{k \in H_k} \widehat{d}_{ik}} & \Omega^f_j \in H_j; \\ \widehat{d}_{ij} & \Omega^f_j \in I_j \text{ and } \Omega^f_j \notin H_j; \\ 0 & \Omega^f_j \notin I_j; \end{cases} \tag{4.25}$$

   The last sub-function in Equation (A.21) clears increments of fine-scale volumes as they get to the boundaries of the support region. To avoid losing partition of unity, the first sub-function of Equation A.21 redistributed this increment among all other support regions that share these fine-scale volumes. In essence, while the second sub-function spreads the influence inside each Prolongation Operator column $(\underline{P}_{op})^{\eta}_{j}$, the last sub-function restricts this influence to fine-scale volumes contained in each support region and while the first spreads this lost influence by normalizing the Prolongation Operator rows $(\underline{P}_{op})^{\eta}_{i}$ of fine-scale volumes that belong to the Global Support Boundary. Thus, sub-functions 1 and 3 are a sort of Reduce Boundary conditions. It is worth remembering that the primal coarse cell center (x^P) by definition does not belong to no support region nor any boundary region, therefore the values attributed by Equation 4.22 remain constant as Equation (A.21 is incapable modifying the value of these volumes. In other words, the centers take 1 for values on the Prolongation Operator Column $(\underline{P}_{op})^{\eta}_{x^P j}$ associated with the primal coarse volume and 0 otherwise.

3. The Prolongation Operator is updated using the modified increment.

$$(\underline{P}_{op})^{\eta+1}_{j} = (\underline{P}_{op})^{\eta}_{j} + d_j \tag{4.26}$$

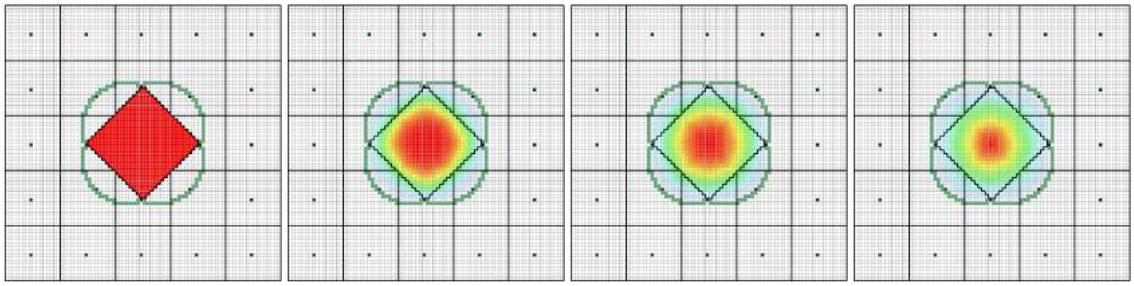
4. The local error is defined for volumes not in the Global Support of the Boundary Region.

$$e_j = \max_i(\|\widehat{d}_{ij}\|), \quad i \notin G \tag{4.27}$$

5. Check for convergence. If $\|e_j\|_\infty > tol$ the current $(\underline{P}_{op})^{\eta}_{j}$ is fed to Step 1, otherwise set $(\underline{P}_{op}) = (\underline{P}_{op})^{\eta}_{j}$.

The Multiscale Restricted Smoothed Basis is named after the process reported above. Figure (33) to (36) illustrate how the initial indicator function is smoothed restricted to the corresponding support region. In both cases, they assume hat-shaped functions bounded by the support region.

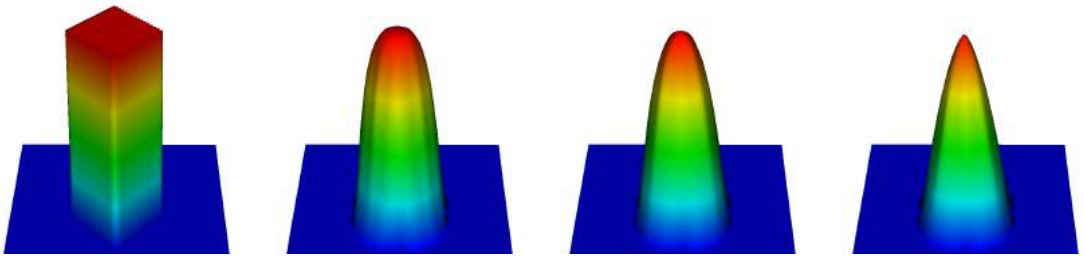
Figure 33 – Illustration on how the basis function spreads its influence inside its associated support region calculated for at homogeneous permeability field.



(a) Initial guess. (b) 41 iterations (c) 81 iterations. (d) After converging.

Source: Author.

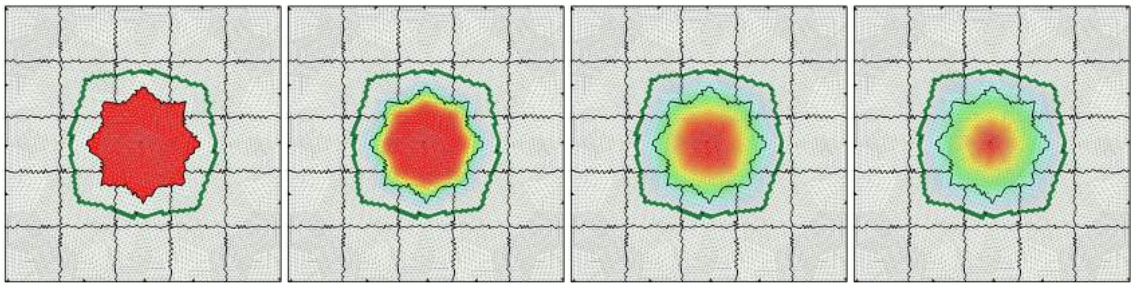
Figure 34 – Three-dimensional view on the basis function of Figure 33 being smoothed while restricted to its support region.



(a) Initial guess. (b) 41 iterations. (c) 81 iterations. (d) After converging.

Source: Author.

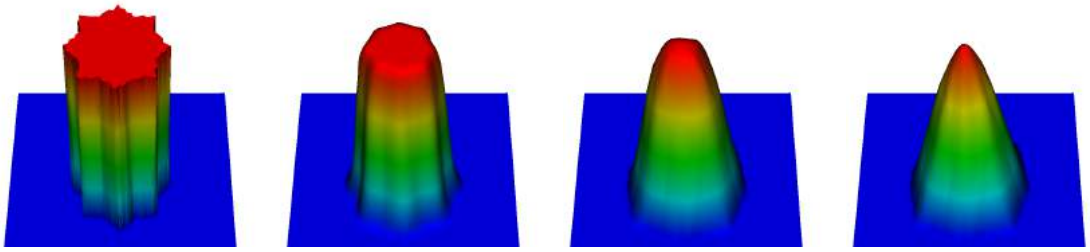
Figure 35 – Illustration showing the basis function in a unstructured grid converging limited to its support region unstructured fine-scale grid.



(a) Initial guess. (b) 31 iterations. (c) 171 iterations. (d) After converging.

Source: Author.

Figure 36 – Three-dimensional view on the basis function of Figure 35 that shows a star-shaped initial guess iterating into a hat-function type restricted to its support region.



(a) Initial guess. (b) 31 iterations. (c) 171 iterations. (d) After converging.

Source: Author.

5 RESULTS

In this chapter we present the results obtained using the MsCV framework implemented on Matlab 2014 (MATLAB, 2014). For the sake of simplicity the results are grouped in accordance to the type of flow studied, one or two-phase flows. All simulations were performed in accordance with the numerical formulation described on Chapter 3. It is worth noting that reference solution stands for the solution obtained by performing a direct simulation on the fine-scale or the analytical solution when available.

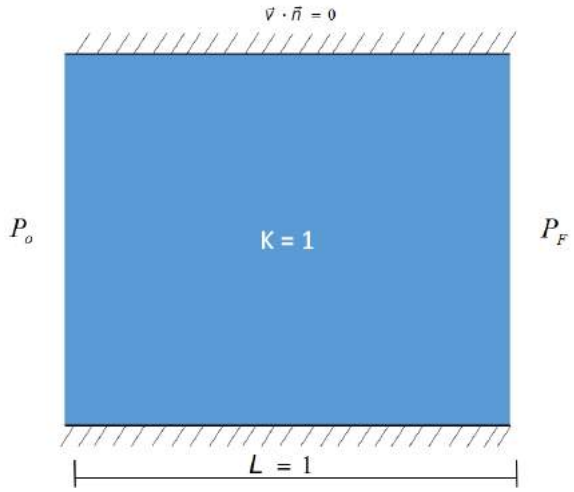
5.1 One-Phase Flow Results

The results presented on this subsection use one-phase flow examples to better understand the behavior of the MsCV framework and its components. The first example was devised to verify how MsCV impacts the piece-wise linear property of the MPFA-D, the second example is a benchmark of Metis and geometric partitioner generated coarse-grids, the third example consists on a test showing the possibilities of using an adaptable coarse-grid and the last example uses a knowingly difficult highly heterogeneous permeability field to evaluate the accuracy of the MsCV framework.

5.1.1 MsCV Piecewise Linearity Test

Among the many features desired for numerical schemes we highlight the ability to produce solutions that are piece-wise linear. Gao and Wu (2010) devised the MPFA-D under specific assumptions in order to guarantee this attribute. Therefore, it is worth testing how the multiscale framework affects this ability. To do so, we use the classical Flow Channel problem (See Figure 37) in a homogeneous squared domain $[0\ 1]^2$ in which the top and the bottom are submitted to null flux condition and the left and right side to prescribed pressure respectively defined as 1 and 0.

Figure 37 – Flow-channel domain representation.



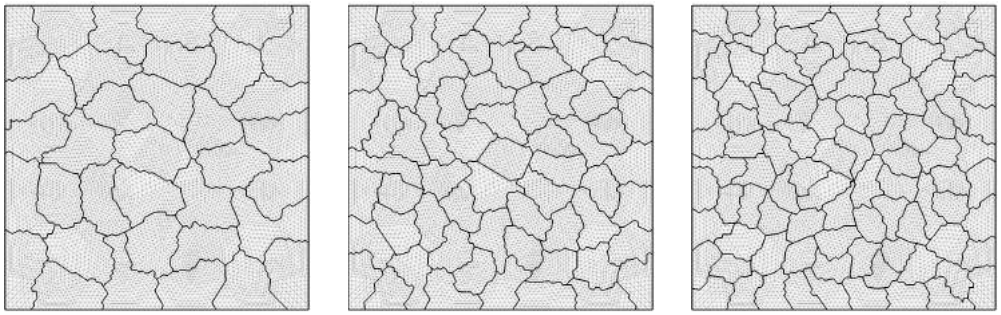
Source: Author.

This problem has a quite simple analytical solution expressed by the following equation:

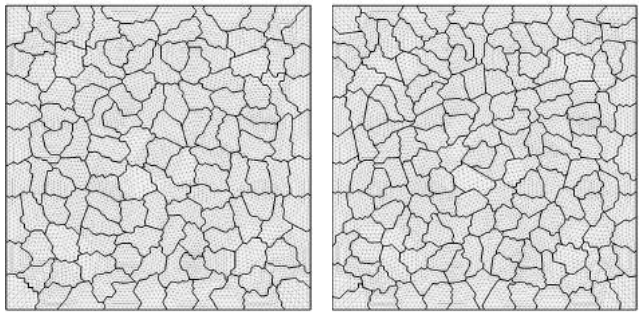
$$p(x) = -x + 1 \tag{5.1}$$

The experiment we propose solves this problem using the multiscale framework in high-resolution grid containing 14,592 triangular elements coarsened using the geometric partitioner and the Metis algorithm producing each 5 coarse-scale meshes (See Fig.38 and Figure 39).

Figure 38 – Coarse grids generated using Metis.



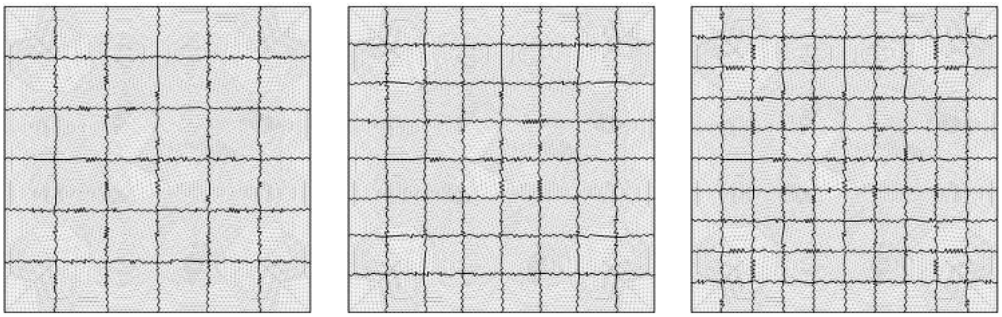
(a) 35 coarse volumes. (b) 65 coarse volumes. (c) 95 coarse volumes.



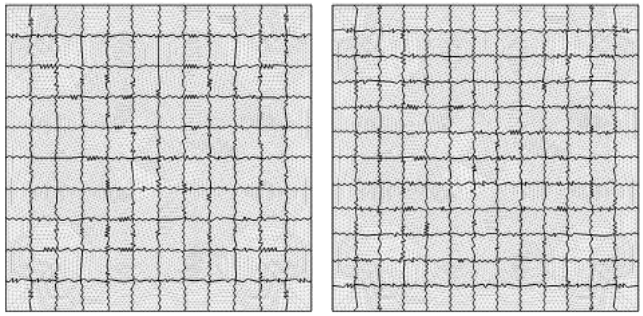
(d) 125 coarse volumes. (e) 155 coarse volumes.

Source: Author.

Figure 39 – Coarse grids generated using Geometric Partitioner.



(a) 36 coarse volumes. (b) 64 coarse volumes. (c) 100 coarse volumes.



(d) 120 coarse volumes. (e) 156 coarse volumes.

Source: Author.

This expected solution to this problem, as defined by Equation (5.1), is a linear function that connects the prescribed pressure values of the left to that on right of the physical domain. As we have previously mentioned, the MPFA-D was conceived to ensure piece-wise linear solutions. It is worth noting that for this configuration the direct simulation on the fine-scale converged to the analytical solution. The MsCV in turn produced solution that were coase-scale dependent and oscillated around the analytical solution.

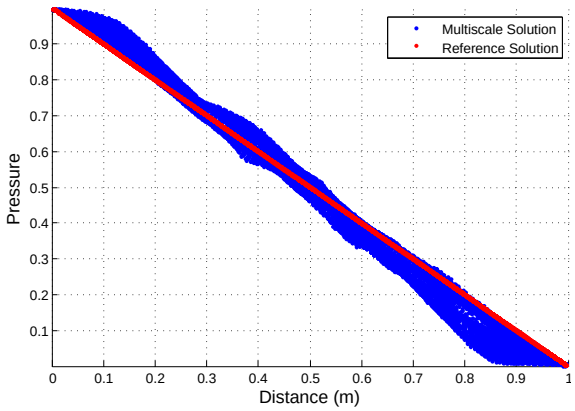
Figures 40 and 41 showed that no matter the number of coarse volumes Metis is clearly more oscillatory than the Geometric Partitioner. Additionally, it was easy to notice that the MsCV could not preserve the piece-wise linear property in all coarse-grids tested. Nonetheless Table 2 shows that with a proper coarsening the average pressure error can drop close to 1%. Finalt wly, it was clear that the geometric partitioner produced the better solution in comparison to the Metis. Check Table 1 and 2. It is worth pointing out that the 95 coarse volumes grid created using Metis produced pressure values so oscillatory that infinity norm $|p|_\infty$ peaked at 76.1097%.

We speculate that as the coarse volumes are highly unstructured with different shapes and sizes, the corresponding support regions also vary in dimension and quality. As a consequence, the domain is unevenly covered with support regions what leads to parts of the domain influencing too much the coarse pressure values over others. Thus, an small error on the coarse pressure are amplified by the prolongation operator causing the presented oscillations.

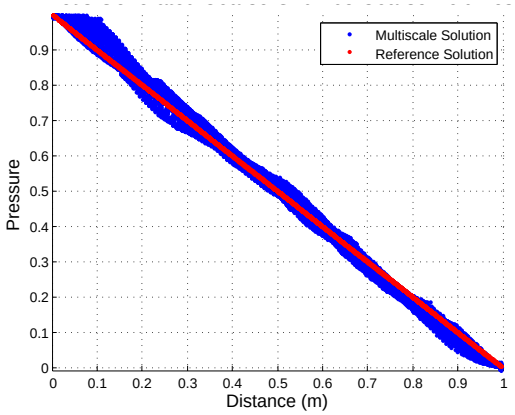
It is worth point out that these errors bear little influence on the overall quality of the two-phase flow solutions. Firstly, because the saturation field relies on the velocity fields that in turn rely on the pressure gradient. Secondly, these error are small in comparison to those introduced by highly heterogeneous and/or anisotropic medium as the examples in the two-phase flow subsection show. In these cases, the simulation using the Geometric Partitioner produced solutions with errors as small as MsFV, which is also piece-wise linear.

The next results corroborate with this proposition. Moreover, there are state of the art smoothers that are capable of significantly improving the pressure field with only a few iterations. Hajibeygi and Jenny (2011) develop an algorithm that improves the approximation of the basis functions iteratively up to a given tolerance, Barbosa (2017) recalculates the pressure field solving the Equation 2.9 on each primal coarse volume bounded by Dirichlet BC using the pressure values obtained from Equation 4.14.

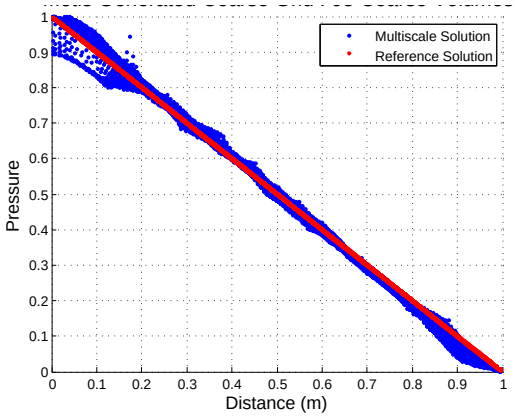
Figure 40 – Comparison between the multiscale and reference solution obtained using coarse-grids generated on Metis.



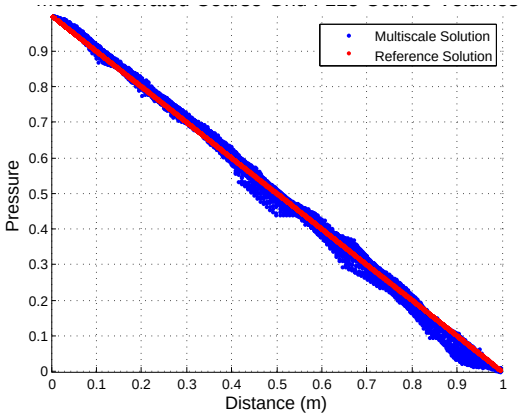
(a) 35 coarse volumes.



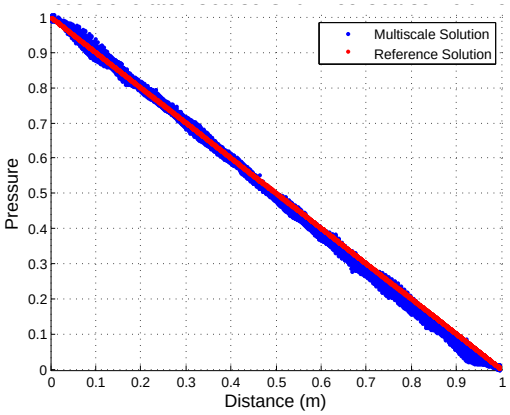
(b) 65 coarse volumes.



(c) 95 coarse volumes.



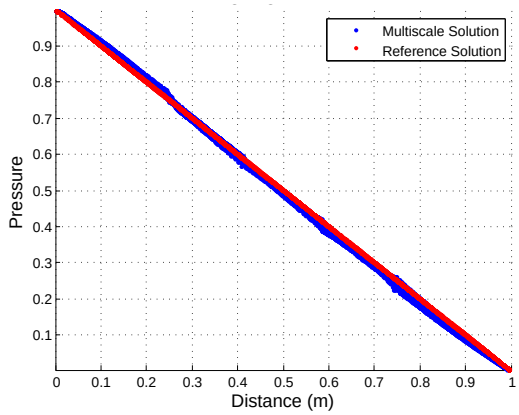
(d) 125 coarse volumes.



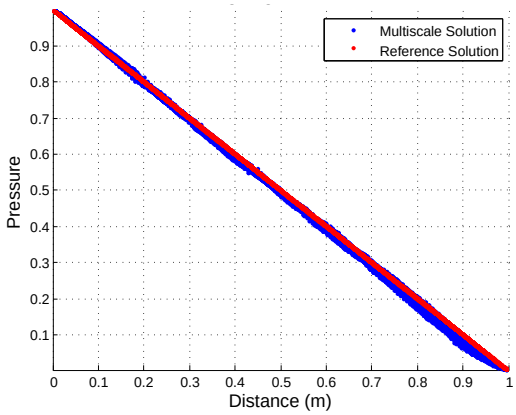
(e) 155 coarse volumes.

Source: Author.

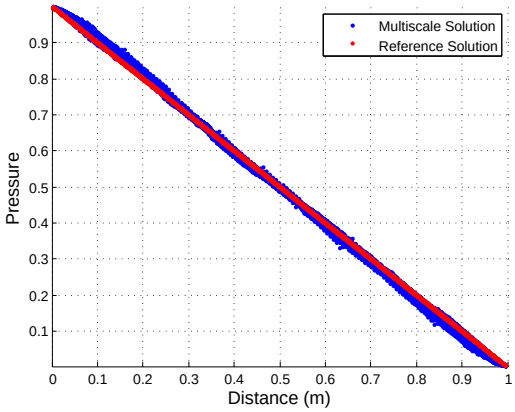
Figure 41 – Comparison between the multiscale and reference solution obtained using coarse-grids generated on our geometric partitioner.



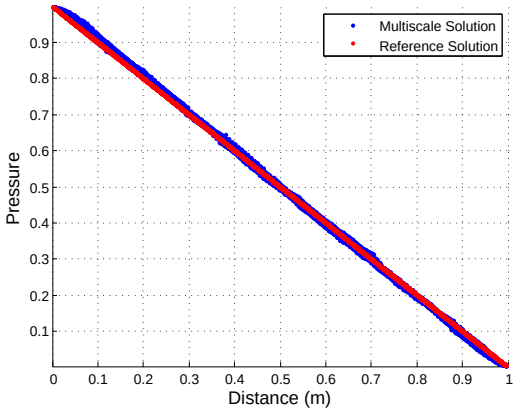
(a) 36 coarse volumes.



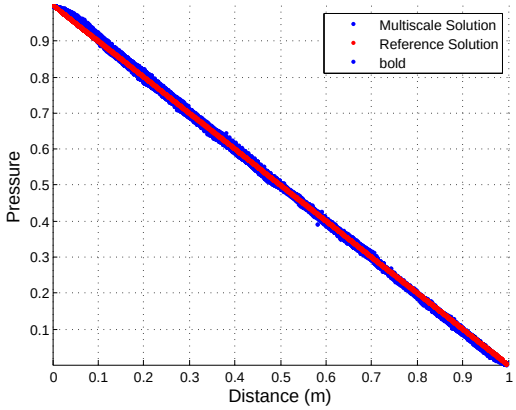
(b) 64 coarse volumes.



(c) 100 coarse volumes.



(d) 120 coarse volumes.



(e) 156 coarse volumes.

Source: Author.

Finally, we have noticed a clear trend presented on the Tables 1 and 2 and on the Figures 40 and 41 showed that as the coarse grids become more well behaved the multiscale solution converges to the reference solution. To test this proposition, we performed a simulation using a 9x9 structured fine-scale grid on a 3x3 structured coarse-scale grid. The simulation was performed using the center of coarse volumes

Table 1 – MsCV piecewise linearity test errors of the Metis grids.

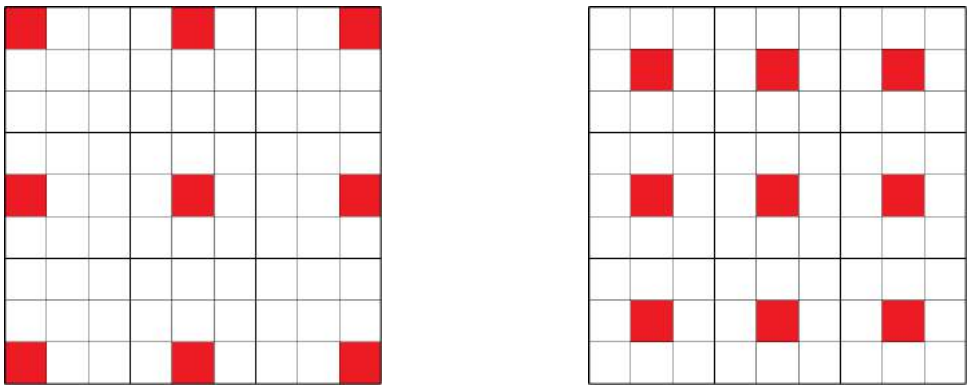
# of Coarse Volumes	$\|p\|_2$	$\|p\|_\infty$	Coarsening Algorithm
35	6.5025%	13.3116%	Metis
65	4.1776%	9.5648%	Metis
95	3.1206%	76.1097%	Metis
125	2.4210%	6.9305%	Metis
155	2.7314%	5.5654%	Metis

Table 2 – MsCV piecewise linearity test errors of the geometric partitioner grids.

# of Coarse Volumes	$\|p\|_2$	$\|p\|_\infty$	Coarsening Algorithm
36	2.0166%	3.4052%	Geometric Partitioner
64	1.5651%	4.219%	Geometric Partitioner
100	1.8609%	4.215%	Geometric Partitioner
120	1.3986%	2.784%	Geometric Partitioner
156	1.0303%	2.9747%	Geometric Partitioner

proposed by Moyner and Lie (2015), in which the coarse centers coincide with the centroid of each coarse volume for quadrilateral structured grids, and the coarse centers proposed by Barbosa (2017) that suggests relocating the center of primal coarse volumes lying on the boundary of the domain to the fine-scale volume closest to the intersection between the coarse cell surface and the physical domain (See Figure 42). The pressure solution presented on Figure 43 points out that using well behaved structured grids is not sufficient condition to ensure piece-wise linearity as this property is also depend of a proper choice for primal coarse center. The coarse center type suggested byMoyner and Lie (2015) did not converge to the reference solution, however, the relocated coarse center proposed by Moyner and Lie (2015) did.

Figure 42 – Primal coarse centers used to check piece-wise linearity on structured grids.

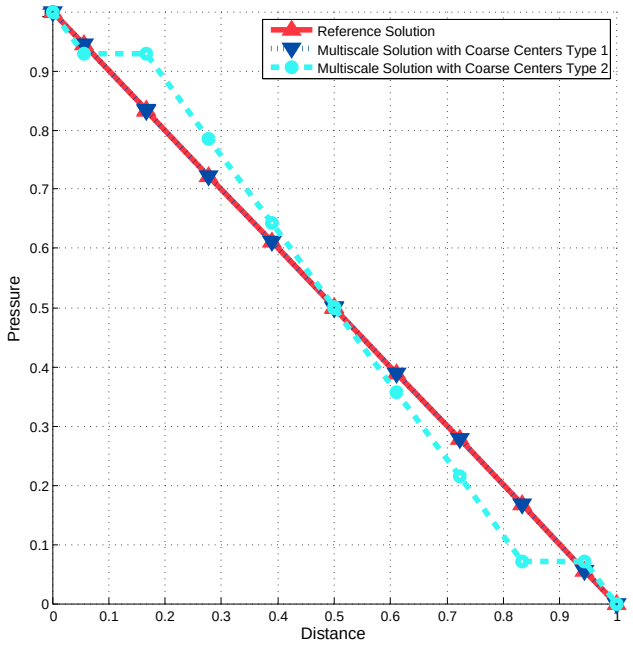


(a) Primal coarse center proposed by Barbosa (2017) (Center type 1).

(b) Primal coarse center proposed by Moyner and Lie (2015) (Center type 2).

Source: Author.

Figure 43 – Comparison between the multiscale solution for each type of center and reference solution.

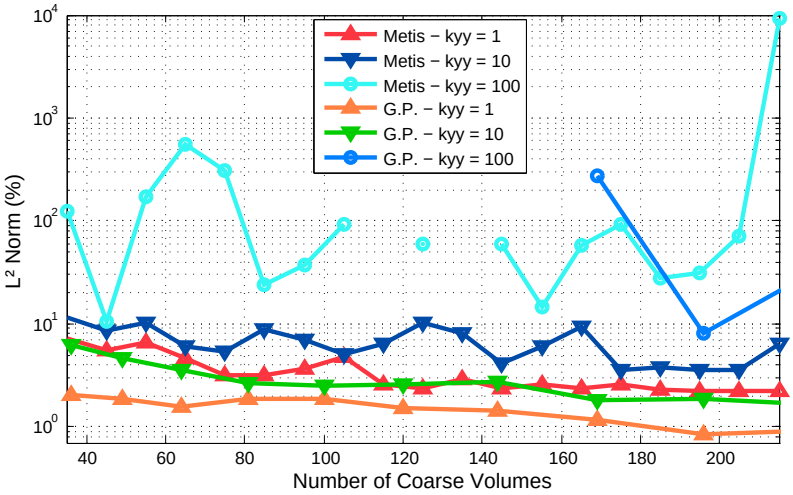


Source: Author.

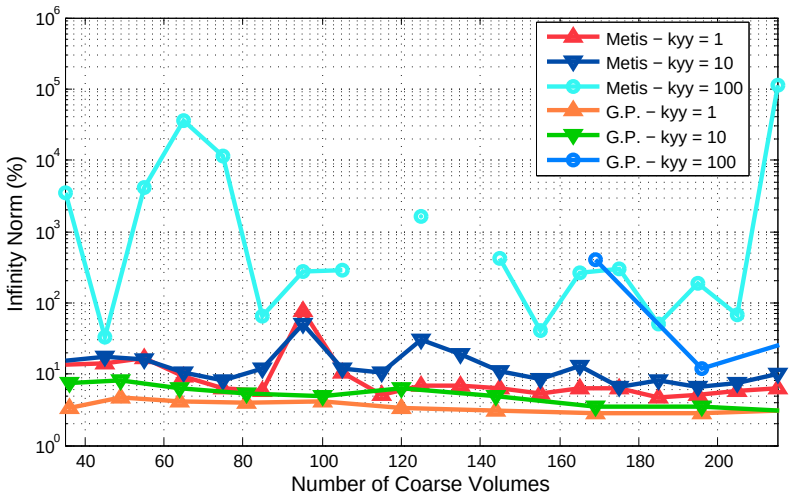
5.1.2 A benchmark of coarse-grids generated using the Geometric Partitioner and Metis

The need to better understand the results of the last subsection lead us to propose a numerical experiment to help determine in which cases the Geometric Partitioner produces better results than Metis. We have chosen to use once more the Flow-Channel problem keeping the underlying fine-scale grid with 14,592 elements. Additionally, we have created 20 coarse grids using Metis in which the coarse volumes range from 35 up to 225 and 13 coarse grids using the Geometric Partitioner counterpart with volumes ranging from 36 to 256. For each generated coarse grid we performed a simulation varying the K_{yy} component of the permeability tensor taking the values 1, 10 and 100. The results obtained were compiled into the following 2 charts presented on Figure 44.

Figure 44 – Error of the coarse grids generated using Metis and the Geometric Partitioner (GP).



(a) L2 Norm Comparison.



(b) Infinity Norm Comparison.

Source: Author.

The first feature that call our attention was the rather stable and low error obtained using the grids created using the Geometric Partitioner. It is noteworthy how smooth the L2 and Infinity norms curves for $K_{yy} = 1$ and $K_{yy} = 10$. In addition, they also produce a downward trend in both studied norms. Meanwhile, Metis produced oscillatory error curves that showed a clear downward trend only for K_{yy} in the L_2 norm. Also, the L2 and Infinity norm values on the Metis curves were considerably higher than the Geometric Partitioner counterpart.

The high anisotropic case ($K_{yy} = 100$) in turn produced extremely oscillatory results for both partitioner. The gaps presented on these curves represent coarse-grids in which the error become so high that multiscale framework beard no solution. Thus, the Geometric Partitioner was not capable of producing a stable solution for the first 10

coarse-grids generated while Metis did not produced a solution only in 2 cases.

To understand why of this unexpected result we need to remember that the prolongation operator is calculated neglecting the normal component of the flux flowing outside the support region. Moreover, the K_{yy} component of the permeability tensor may induces a trend to flow 100 times greater on the Y direction. Thus, as the Geometric Partitioner generates well behaved coarse-grids in which the faces of the coarse volumes tend to stay parallel to the main directions, the reduced boundary conditions destroy the Y component of flow depriving the support region of the physics of the problem.

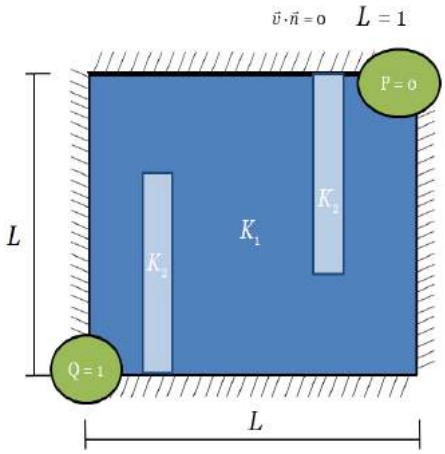
On the other hand, the uneven and aleatory shapes of the faces of the coarse volumes generated on the Metis produces contours not even close to the main components of the flow, which means that the impact of reduced boundary conditions are scaled down. In other words, the surface of the coarse-grid volumes are better aligned to the preferential direction of the permeability tensor.

5.1.3 Using the Geometric Partitioner to create an adaptive grid

The MsCV inherits from the MsRSB the ability to work with truly unstructured coarse-grids. To take advantage of this property, the Geometric Partitioner incorporate routines capable of generating non-conforming lower-resolution grids as thoroughly shown on the last chapter. For this reason, we propose a simple numerical experiment to show the impact of using an adaptive mesh on the solution. The experiment consists on an unitary domain $\Omega = [0\ \ 1] \times [0\ \ 1]$ in a 1/4 of five spot configuration, in which, flux is prescribed ($Q_{inj} = 1$) at the injector well and pressure is prescribed ($p_{prod} = 0$) at the producer well. The medium is isotropic and heterogeneous and it consists in two transverse channels as depicted on Figure 45. The permeability tensors of this example are given by the following expression:

$$K_1 = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \quad \text{and} \quad K_2 = \begin{bmatrix} 1000 & 0 \\ 0 & 1000 \end{bmatrix}$$

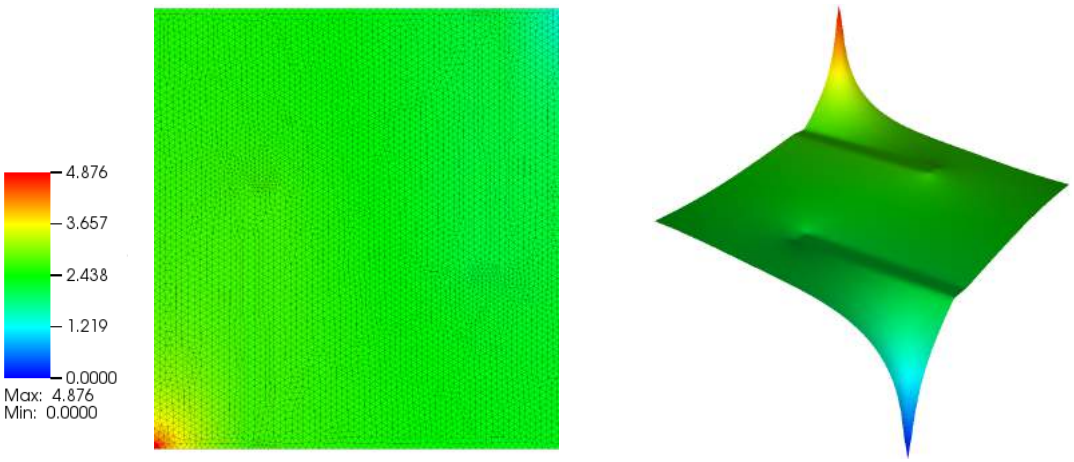
Figure 45 – Homogeneous medium with high permeability channels.



Source: Author.

This problem has been focus of many studies on literature (BARBOSA, 2017; CARVALHO, 2005; JENNY; LEE; TCHELEPI, 2006). Figure 46 presents the pressure field of the reference solution for comparison purposes. Usually, it does not present great challenge to multiscale methods, however when the vertical boundaries of the coarse volumes meet the high-permeability channels, the multiscale solution deteriorates. This happens because the boundary conditions used to uncouple the problem neglect the flux where the permeability tensor induces bigger fluxes.

Figure 46 – Pressure field of the reference solution of problem.



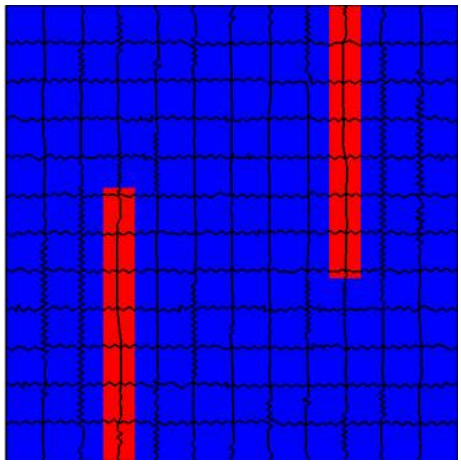
(a) Two-dimensional view of the pressure field.

(b) Three-dimensional view of the pressure field.

Source: Author.

To replicate this studies, we generated a 12x12 coarse grid upon a fine-scale unstructured grid with 12,018 triangular elements in which the vertical boundary of the coarse volumes falls inside the channels (See Figure 47).

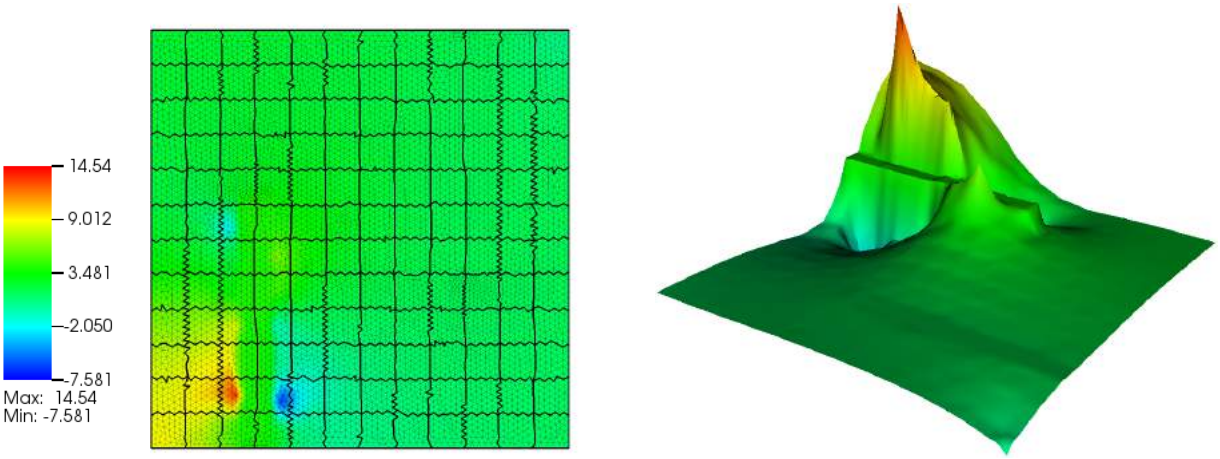
Figure 47 – 12x12 coarse grid on a homogeneous by part medium with high permeability channels.



Source: Author.

Figure 48 shows that, as expected, the MsCV produced non-physical results that qualitatively bear no resemblance to the reference solution.The L2 and Infinity norm of the pressure solution ($|p|_2$ and $|p|_\infty$) calculated for this problem were estimated in 69.52% and 242.12% respectively. Pressure field overshoot the maximum and undershoot minimum values of the reference solution destroying completely the physics of the problem.

Figure 48 – Pressure field of the multiscale solution of problem.



(a) Two-dimensional view of the pressure field.

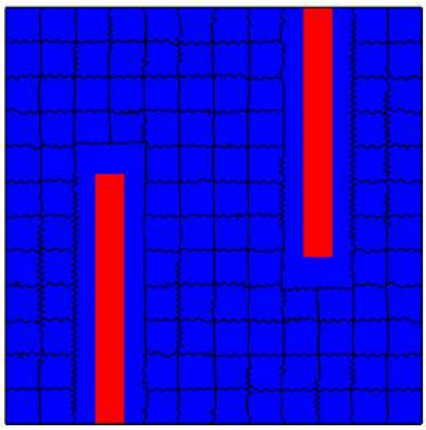
(b) Three-dimensional view of the pressure field.

Source: Author.

To overcome this issue, we used the Geometric Partitioner to tailor a lower-resolution grid respecting the physical properties of the medium. The resulting coarse grid replaces those volumes whose boundaries meet the high permeability regions with

two bigger coarse volumes so that these channels can lie completely inside each coarse block (See Figure 49).

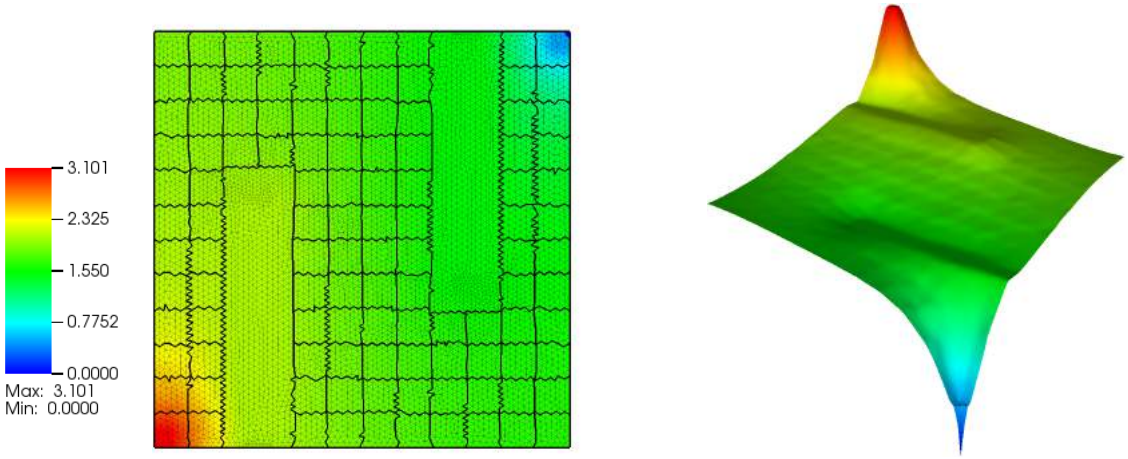
Figure 49 – Non-conforming "12x12" coarse grid whose coarse volumes do not meet regions of high permeability channels.



Source: Author.

The simulation performed on this adaptable coarse grid exceed our expectations. First the L2 and Infinity norm of the pressure solution ($|p|_2$ and p_∞) dropped to respectively 26.10% and 36.45%. In addition, the pressure field did not overshot the maximum or undershoot minimum values of the reference solution. In the overall, the new pressure field was greatly improved in comparison to prior multiscale solution. In conclusion a proper choice of coarse-grid was capable of enabling simulations diminishing errors and respecting the physics of the problem.

Figure 50 – Pressure field of the multiscale solution of problem 5.1.3 on a non-conforming coarse grid.



(a) Two-dimensional view of the pressure field.

(b) Three-dimensional view of the pressure field.

Source: Author.

5.1.4 Highly Heterogeneous and Highly Anisotropic Permeability Field

Let us consider now a one-phase flow in a highly heterogeneous and anisotropic media inside a square domain with a unitary side (CONTRERAS, 2012). The pressure on the boundaries of the domain are set to zero and the permeability tensor is given by the following expression:

$$\underset{\sim}{K} = \begin{bmatrix} y^2 + \epsilon x^2 & -(1-\epsilon)xy \\ -(1-\epsilon)xy & \epsilon y^2 + x^2 \end{bmatrix}, \quad \epsilon = 5x10^{-2} \tag{5.2}$$

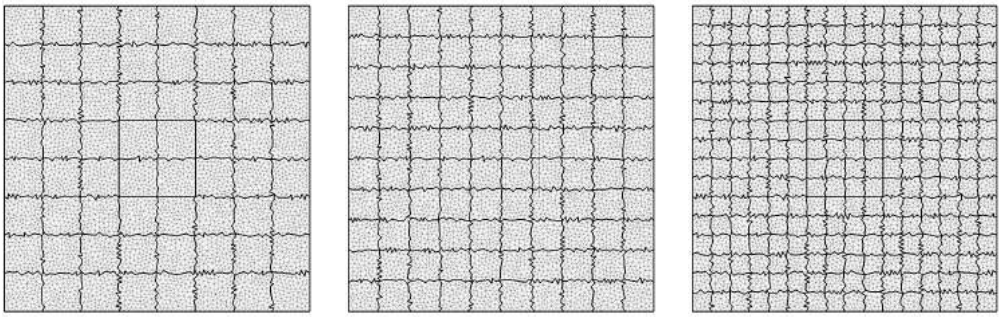
A unitary source term spreads its influence in a smaller square concentric to the domain and is defined as:

$$f(x,y) = \begin{cases} 1 & (x,y) \in \left[\frac{3}{8} \ \frac{5}{8}\right]^2; \\ 0, & (x,y) \notin \left[\frac{3}{8} \ \frac{5}{8}\right]^2 \end{cases} \tag{5.3}$$

Contreras (2012) adapted the problem proposed by Yuan and Sheng (2008) to expose the flaws in the MPFA-D formulation. Contreras showed that the MPFA-D was capable of producing smooth solutions in triangular grids however it failed to respect the Discrete Maximum Principle producing pressure values above and bellow those defined for the boundaries giving birth to non-physical values.

We revisit this problem to understand how the multiscale framework behaves in this case. For this simulation we used for the high-resolution grid a triangular mesh with 10,874 elements. As for the low-resolution grid we use 3 different coarsening ratios, 8x8, 10x10 and 16x16 as illustrated by Figure 51.

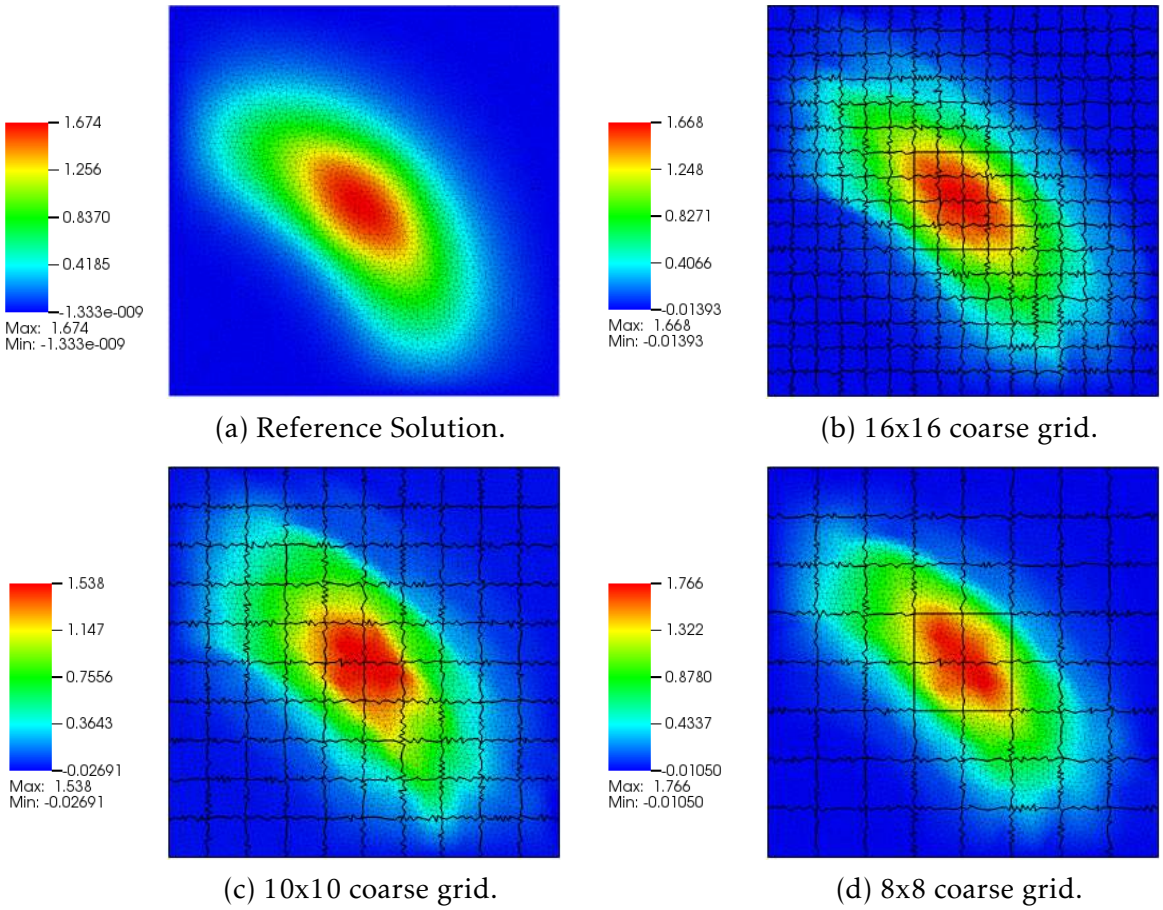
Figure 51 – Illustration on the 3 different coarsening used with the triangular fine-scale mesh with 10,874 elements.



(a) 8x8 coarse grid. (b) 10x10 coarse grid. (c) 16x16 coarse grid.

Source: Author.

Figure 52 – Pressure solution using the 3 coarsening presented on Figure 51.



(a) Reference Solution.

(b) 16x16 coarse grid.

(c) 10x10 coarse grid.

(d) 8x8 coarse grid.

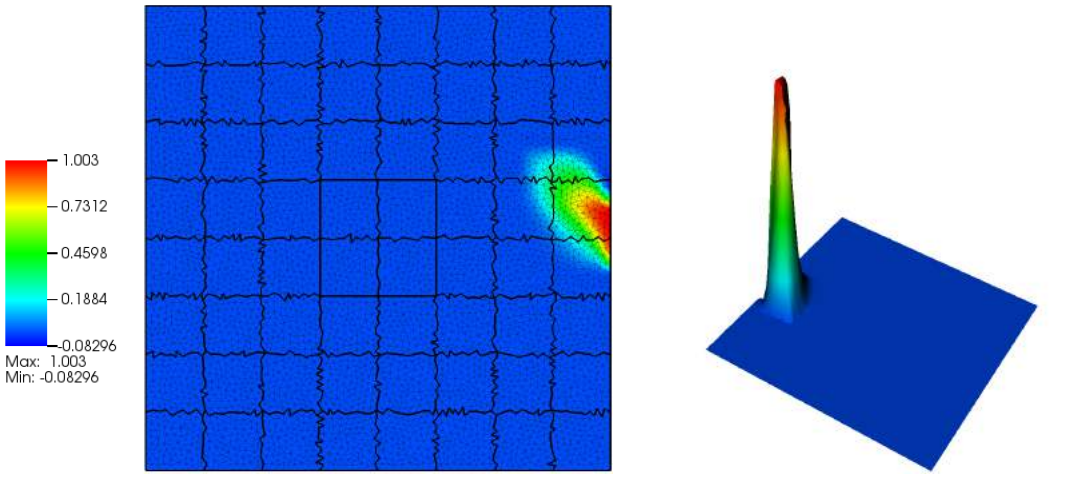
Source: Author.

As shown in Figure 52, not only the MsCV framework replicate the issues presented on the reference solution but it also propagated the errors. In all cases (See Figures 52b to 52d the multiscale solution presented values bellow the minimum value of the reference solution. This is expected, as we understand that the reference solution bounds the quality of the multiscale solution. In other words, if no errors were introduced during the multiscale step the best solution to be obtained would be the reference solution.

As we check the Prolongation Operator we see that the MPFA-D solver used in the multiscale framework produced weights that respected the partition of unity but violated the discrete maximum principle producing weights above 1 and bellow 0. Figure 53 illustrates the support region of a coarse cell on the 8x8 coarse grid lying on the boundary of the domain. Note that in this case the limits of the weights were extrapolated thus giving birth to normalized pressure values with no physical meaning. It is worth noting that many other volumes on the boundary in all coarse grids simulated also presented this behavior. This being said, we speculate that as multiple support regions violate the maximum principle and as the multiscale solution is defined as the

prolongation operator times the coarse-cell pressure vector, this causes the differences of the maximum and minimum value pointed by Figure 52.

Figure 53 – Support region of a coarse cell on the boundary producing values beyond 1 and bellow 0.



(a) Two-dimensional view. (b) Three-dimensional view.

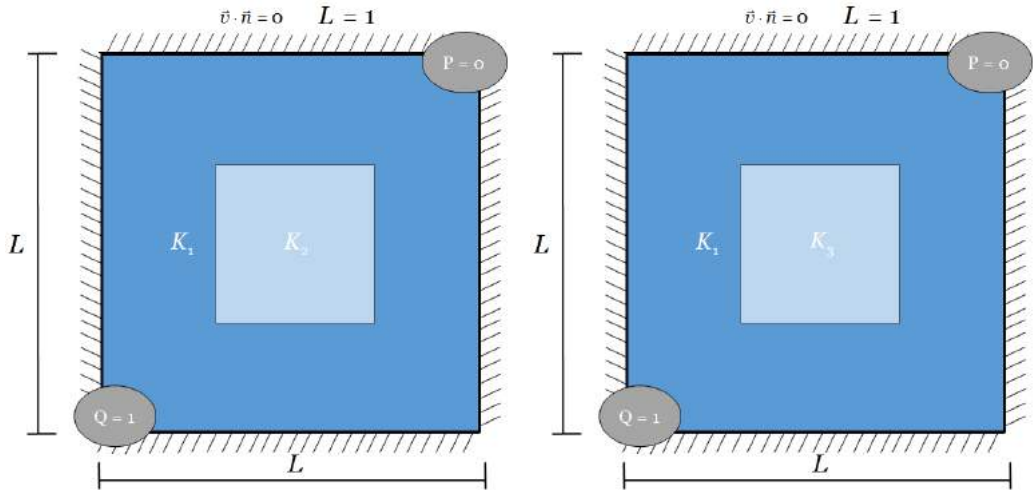
Source: Author.

5.2 Two-Phase Flow Results

In this section we compile the results considering incompressible water-oil flow. All simulation performed in this subsection uses IMPES like approach described on Chapter 3. For the first three examples presented in this subsection we consider a unitary domain $\Omega = [0\ \ 1] \times [0\ \ 1]$ in a 1/4 of five spot configuration, in which, flux is prescribed ($Q_{inj} = 1$) at the injector well and pressure is prescribed ($p_{prod} = 0$) at the producer well. Additionally, water is being injected at the injection well ($S_w = 1$) in a field initially saturated by oil only ($S_o = 1$ or $S_w = 0$). Moreover, the permeability tensors used in these examples are defined by the following expression:

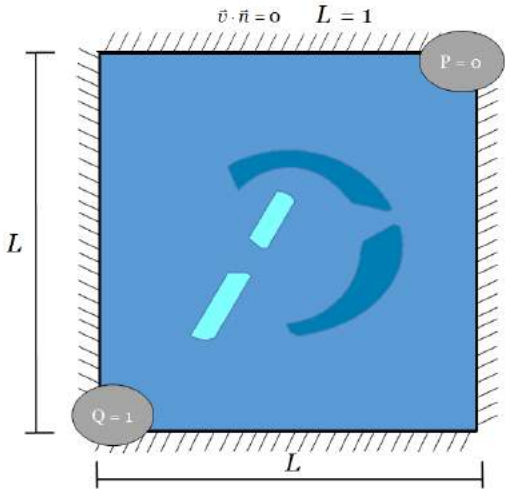
$$K_1 = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \quad K_2 = \begin{bmatrix} \frac{1}{1,000} & 0 \\ 0 & \frac{1}{1,000} \end{bmatrix} \quad \text{and} \quad K_3 = \begin{bmatrix} 1,000 & 0 \\ 0 & 1,000 \end{bmatrix}$$

Figure 54 – Permeability fields of the first three two-phase flow examples.



(a) Example 5.2.1.

(b) Example 5.2.2.



(c) Example 5.2.3.

Source: Author.

5.2.1 Two-Phase Flow in 1/4 of Five Spot with a Central Low Permeability Barrier

In this context, our first example (See Fig. 54*a*) consists in a low permeability zone shaped as a square with size (L = 0.5) and concentric with the unitary domain. In this example, the simulation was performed on a fine-scale grid containing 2,432 volumes and on a coarse-scale 7x7 grid generated using the Geometric Partitioner. Under this setting, we were capable of obtaining results that qualitatively remained very close to the reference solution (See Fig. 55). Notice that the multiscale solution was capable of retaining most of the important features for the flow. These results are supported by the curves in Fig. 58a that show almost identical cumulative oil plots. Nonetheless, the multiscale solution has anticipated the breakthrough. Moreover,

Table 3 shows that the average error on the saturation field is as low as $\|s\|_2 = 1.89\%$ at a PVI = 1.

5.2.2 Two-Phase Flow in 1/4 of Five Spot with a Central High Permeability Channel

Our second example (See Fig. 54*b*) inherits the same geometry, configuration and fine-scale mesh described on Section 5.2.1. Nevertheless, the central barrier is replaced by a high permeability zone in which the permeability tensor is defined as K_3. The simulation in this example was performed on coarse-scale 9x9 grid generated using the Geometric Partitioner. One more time, the multiscale solution were capable of reproducing the physics of the reservoir. Qualitatively, the results remained very close to the reference solution (See Fig. 56). The curves presented in Figure 58b support these affirmations. It is worth point out that for this case the breakthrough took place at roughly the same time as the reference solution. However, there was a slight but significant difference on the cumulative oil curve in which multiscale solution overproduced. Additionally, the errors presented on Table 3 were quite low. It is worth pointing out a diminishing trend over time where the lowest average error $\|s\|_2 = 4.75\%$ takes place at PVI of 1.

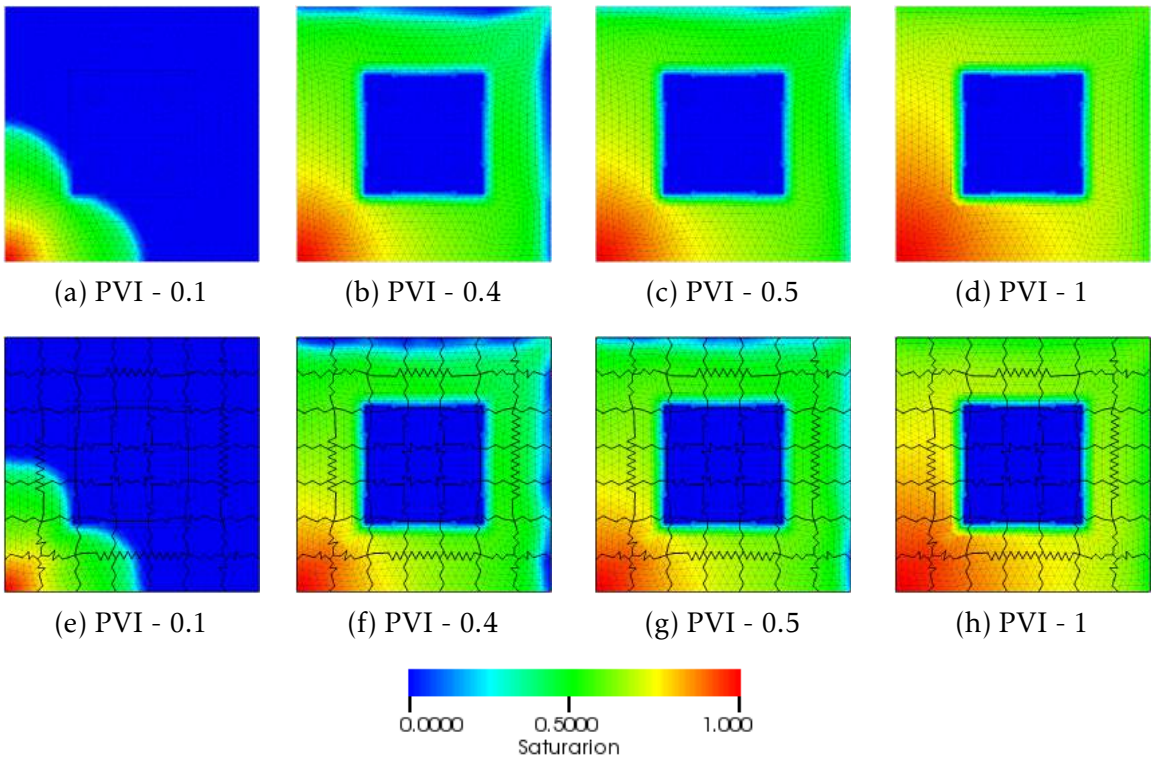
5.2.3 Two-Phase Flow in a 1/4 of Five-Spot with Two Unconnected Channels and Two Curved Barriers

On the third example, we present a geological formation for which classical multiscale methods would have trouble to represent as they would require high-resolution fine-grids to obtain a discrete system capable of preserving the complex features of this geological formation (See Fig. 54*c*). By using our framework, we were capable of running a very accurate simulation with only 3,060 fine-scale volumes and 5x5 coarse volumes. Figure 57 shows that qualitatively the solutions are very close showing that this scheme was capable of preserving the high-resolution features of the flow. The results presented in Fig. 58 support this affirmation by showing very close Cumulative Oil curves. Moreover in this example the breakthrough in the multiscale and in the reference simulation take place at almost the same time. Table 3 also shows a very low average error $\|s\|_2 = 5.28\%$ for a $PVI = 1$.

5.2.4 Flow in a Heterogeneous Reservoir with a Random Permeability Field Using a Fine-Scale Quadrangular Grid

The final example of this thesis consists on incompressible water-oil flow in a unitary domain $\Omega = [0\ \ 1] \times [0\ \ 1]$ in a 1/4 of five spot configuration defined similarly

Figure 55 – Reference and multiscale solution for the saturation field of example 5.2.1.

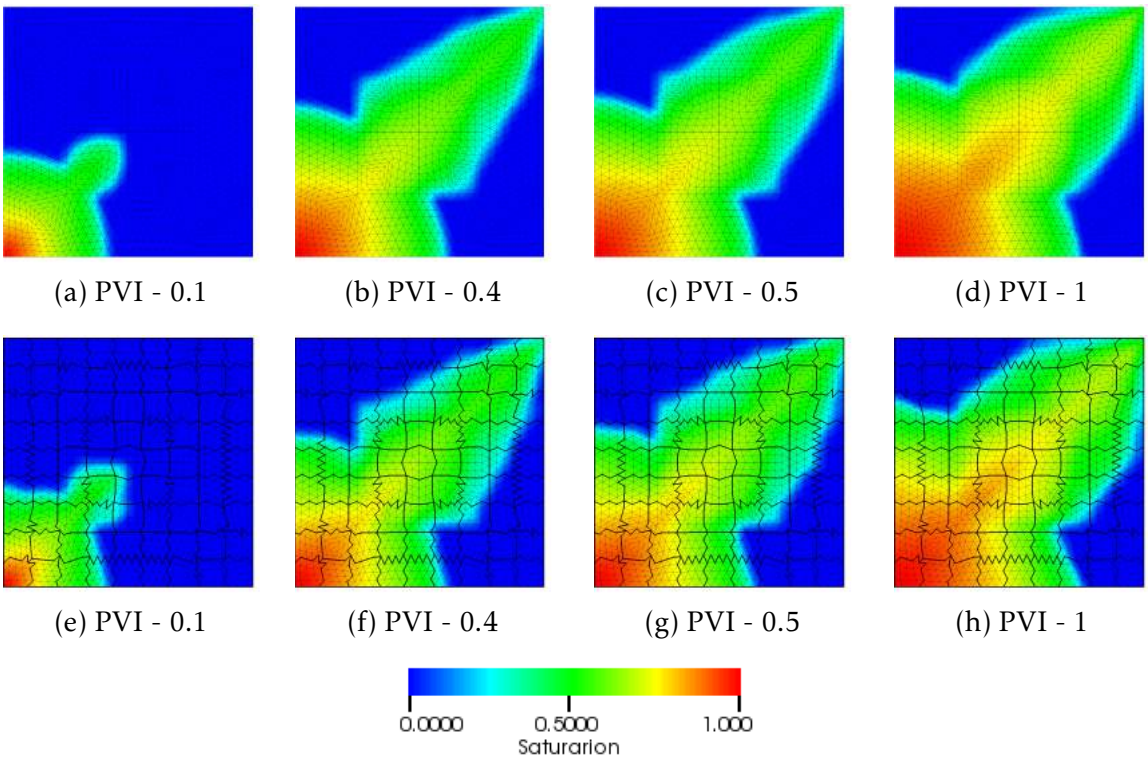


(a) PVI - 0.1 (b) PVI - 0.4 (c) PVI - 0.5 (d) PVI - 1

(e) PVI - 0.1 (f) PVI - 0.4 (g) PVI - 0.5 (h) PVI - 1

Source: Author.

Figure 56 – Reference and multiscale solution for the saturation field of example 5.2.2.

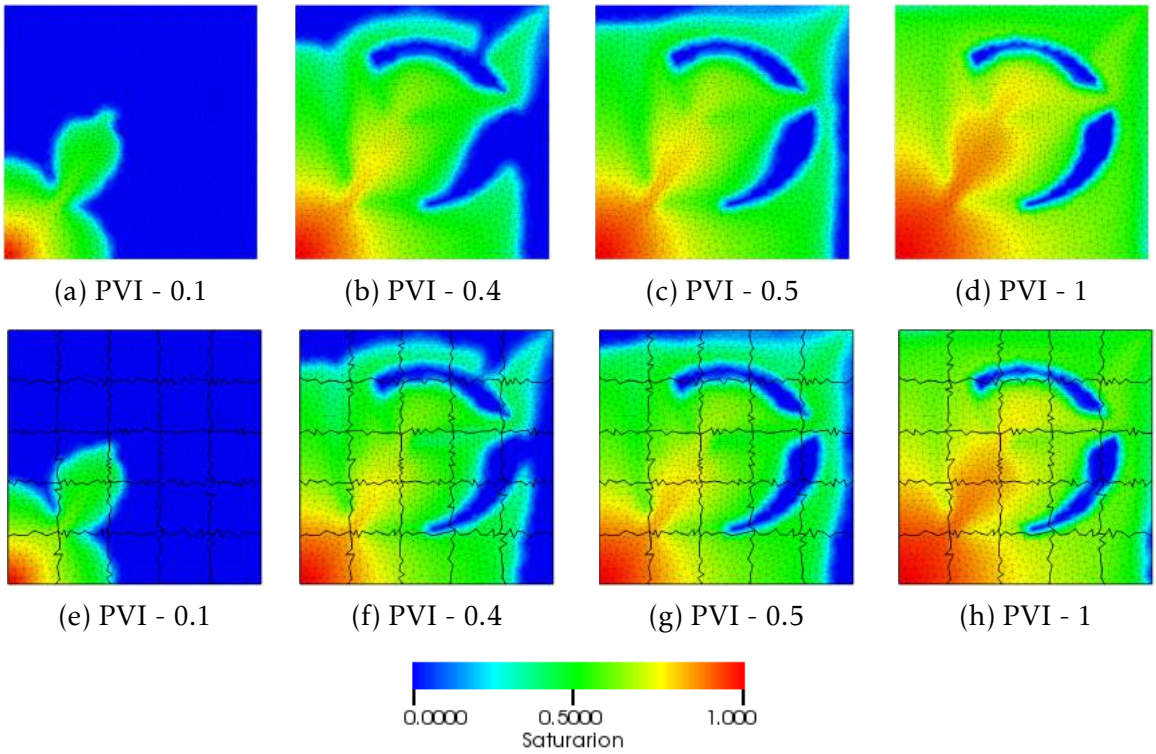


(a) PVI - 0.1 (b) PVI - 0.4 (c) PVI - 0.5 (d) PVI - 1

(e) PVI - 0.1 (f) PVI - 0.4 (g) PVI - 0.5 (h) PVI - 1

Source: Author.

Figure 57 – Reference and multiscale solution for the saturation field for example 5.2.3.

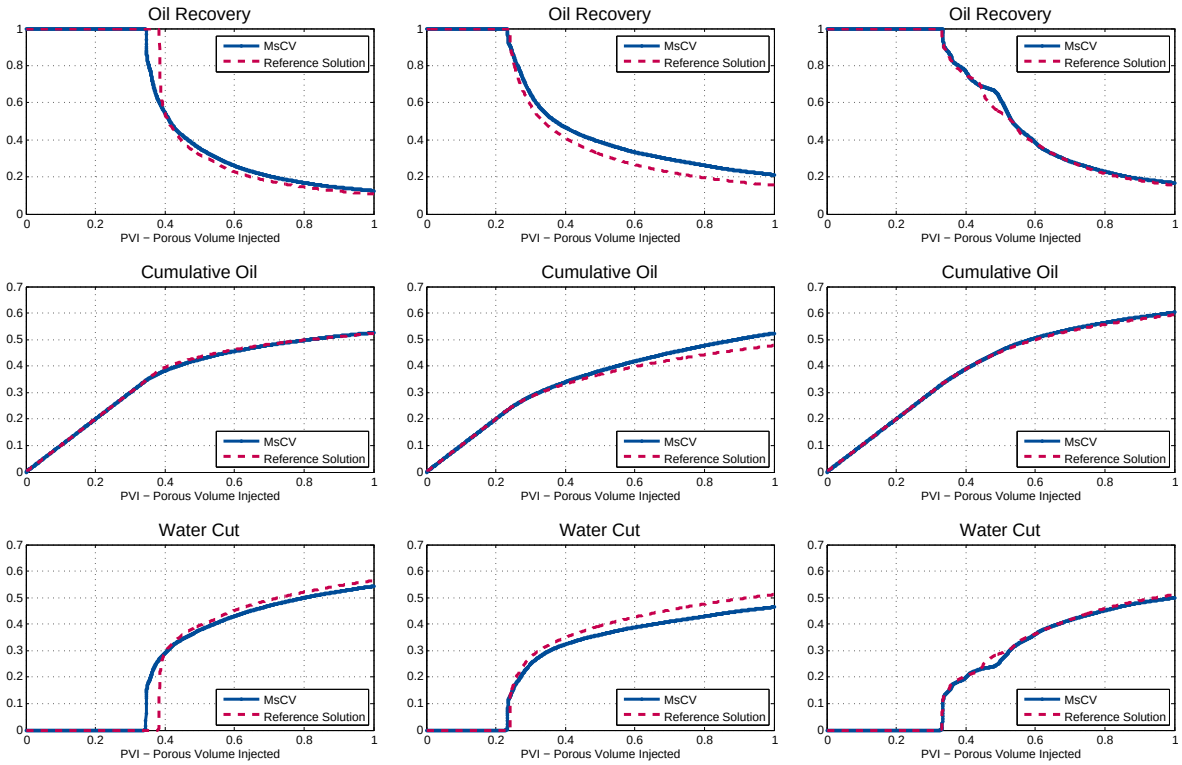


(a) PVI - 0.1 (b) PVI - 0.4 (c) PVI - 0.5 (d) PVI - 1

(e) PVI - 0.1 (f) PVI - 0.4 (g) PVI - 0.5 (h) PVI - 1

Source: Author.

Figure 58 – Production curves of Examples 5.2.1 to 5.2.3 .



(a) Example 5.2.1. (b) Example 5.2.2. (c) Example 5.2.3.

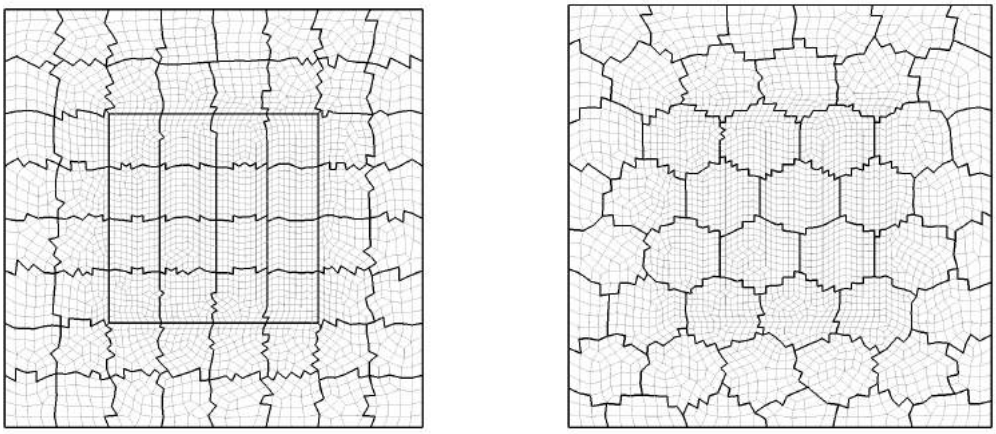
Source: Author.

Table 3 – Average error comparison of Examples 5.2.1 to 5.2.3

PVI	Example 5.2.1 $\|\|s\|\|_2$	Example 5.2.2 $\|\|s\|\|_2$	Example 5.2.3 $\|\|s\|\|_2$
0.1	9.10%	11.30%	14.06%
0.4	6.91%	5.83%	10.86%
0.5	5.24%	5.55%	8.88%
1	1.89%	4.75%	5.28%

as the previous three examples. The permeability of the medium used in this example slightly modifies the permeability tensor proposed by Chueh et al. (2010) that generates random heterogeneous and isotropic fields in order to increase the variability of the medium (See Figure 60). The high-resolution mesh used in this example is a quadrangular grid with 2,795 elements. In addition, we have used the Geometric Partitioner to generate multiple low-resolution meshes in rectangular-shaped and honeycomb-shaped coarse grids. For a matter of comparison we present the two best results of each type, a 8x8 rectangular-shaped grid and with a honeycomb-shaped with 48 coarse volumes as shown in Figure 59.

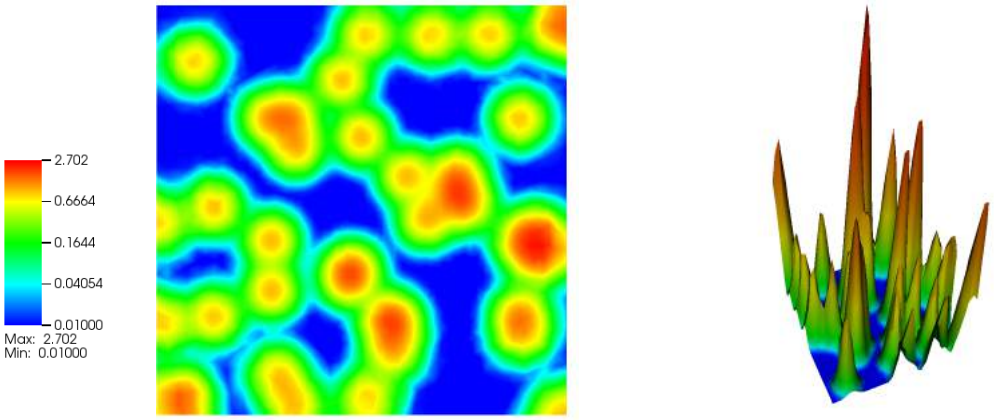
Figure 59 – Fine-scale and Coarse-scale grids of Example 5.2.4.



(a) Rectangular-shaped coarse grid. (b) Honeycomb hexagonal grids.

Source: Author.

Figure 60 – Isotropic and Highly Heterogeneous Permeability Field of Example 5.2.4



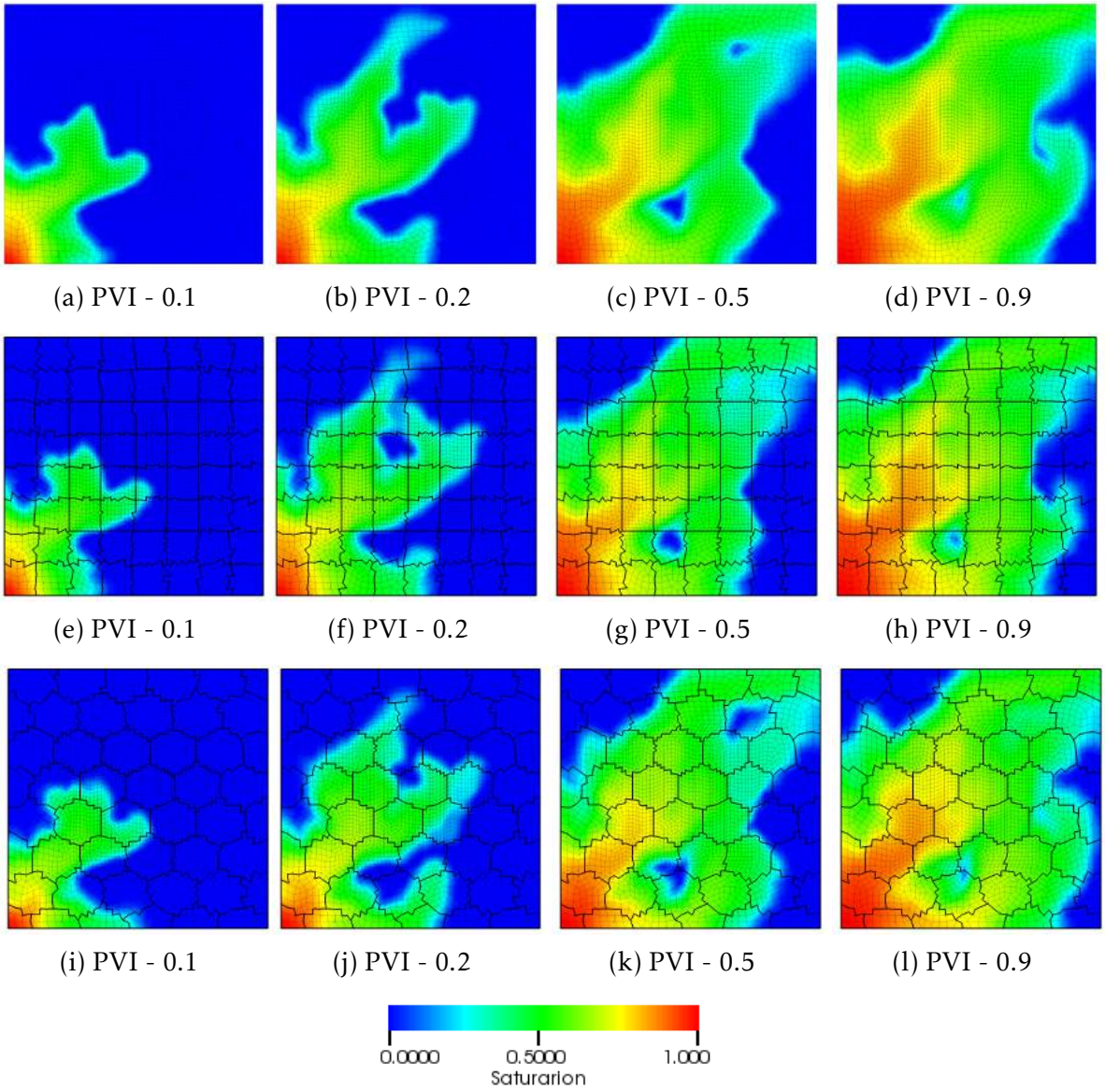
(a) Two-dimensional view. (b) Three-dimensional view.

Source: Author.

As shown on Figure 61, the results were qualitatively very similar. Both grids were capable of retaining the most important features of the flow. However, they also lost some quality on the process. Notice that the bottom-most front of the flow at the PVI 0.2 shows a small phase difference when compared the honeycomb and the reference solution. On the other hand, the same front of the flow on the geometric partitioner has washed-out appearance. Regarding the middle portion of the flow, we point out that both grids produced fronts with a smear look. This phenomena becomes more noticeable from the PVI 0.2 from and on. At a PVI 0.5 the remaining oil accumulates forming two pools. Although the pool located on the bottom-center is decently represented on both multiscale simulations, the one located close to the top right corner is not. The rectangular-shaped seems to produce a flow that is the average of the reference solution meanwhile the honeycomb grid seems to be delayed in comparison to the reference solution.

In general, these solutions were accurate. The rectangular-shaped coarse grid presented a better looking solution at the PVI 0.9 besides that differences on the fingering on the right portion of the flow. Nevertheless, the production curves presented on Figure 5.2.4 point out that the honeycomb grid produces a more accurate solution. Furthermore, Table 4 shows that the norm $\|p\|_2$ of the honeycomb grid is smaller throughout all the simulation, stabilizing at a much lower value, 2.57% in comparison to the 6.88% of the rectangular-shaped grid. On the other hand, the $\|p\|_{\text{inf}}$ is higher on the honeycomb mesh during a considerable time of the simulation. This pattern is only changed at a PVI of 0.9 when this trend is inverted. Moreover, it is worth noting the unsteady bias of the $\|p\|_{\text{inf}}$ on the rectangular mesh.

Figure 61 – Reference, and multiscale solution on a rectangular grid and multiscale solution on the honeycomb grid respectively, for the saturation field for example 5.2.4.



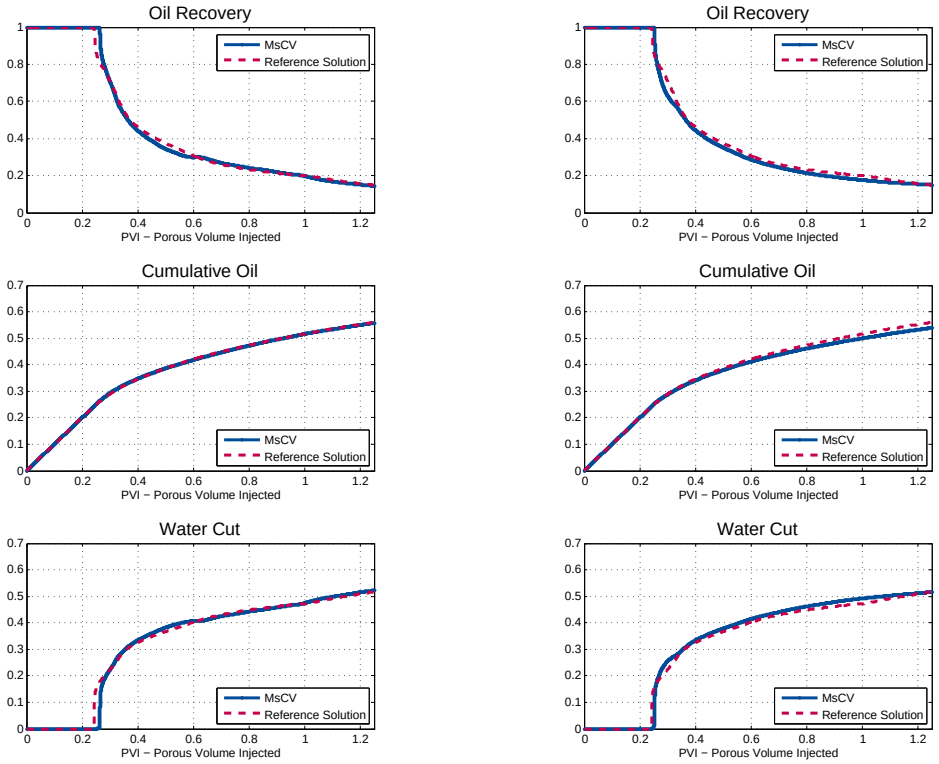
(a) PVI - 0.1 (b) PVI - 0.2 (c) PVI - 0.5 (d) PVI - 0.9

(e) PVI - 0.1 (f) PVI - 0.2 (g) PVI - 0.5 (h) PVI - 0.9

(i) PVI - 0.1 (j) PVI - 0.2 (k) PVI - 0.5 (l) PVI - 0.9

Source: Author.

Figure 62 – Production curves of Example 5.2.4.



(a) Honeycomb mesh production curves. (b) Rectangular mesh production curves.

Source: Author.

Table 4 – Error comparison of the example 5.2.4.

PVI	Honeycomb Mesh		Rectangular Mesh	
	$\|p\|_2$	$\|p\|_\infty$	$\|p\|_2$	$\|p\|_\infty$
0.1	3.03%	19.54%	8.64%	13.19%
0.2	3.32%	22.51%	8.76%	17.43%
0.3	2.99%	19.89%	6.82%	13.45%
0.4	4.47%	15.78%	7.27%	11.82%
0.5	4.58%	13.93%	7.70%	10.54%
0.6	2.82%	11.50%	7.81%	9.86%
0.7	2.53%	9.99%	7.72%	9.75%
0.8	2.57%	9.30%	7.25%	9.66%
0.9	2.57%	8.52%	6.88%	9.91%
1	2.57%	7.84%	6.88%	10.22%

6 CONCLUSIONS AND FURTHER WORK

In this thesis, we modify the multiscale restriction-smoothed basis (MsRSB) replacing the two-point flux approximation (TPFA) by a non-orthodox multi-point flux approximation with a diamond stencil (MPFA-D) to create the multiscale control-volumed (MsCV) framework that allows the simulation on unstructured grids on the fine and coarse scales.

Additionally, we have developed a geometrical coarsening algorithm that creates lower-resolution grids in which the users can generate coarse volumes are capable of adapting themselves according to the underlying geological features of the media. As the examples have shown, this framework loses the piecewise linearity property intrinsic to the MPFA-D, however this does not seem to have a great impact on the two-phase flow solutions.

The routines developed to perform a geometric partition of the high-resolution grid outperformed Metis in almost all cases analyzed. Furthermore, the ability to generate adaptable- coarse grids showed prominent results. In general, the framework created was capable of preserving high-resolution features of the flow even in highly heterogeneous and mildly anisotropic medium.

Moreover, the ability to work with truly unstructured fine-scale grids can improve the discretization of complex geological formations enabling simulation on fine meshes with a significantly lower number of control volume in comparison to the structured grid counterparts.

In conclusion, the framework we have presented can produce accurate solutions at reasonably computational costs on the simulation of two-phase flow in heterogeneous and anisotropic porous media using general unstructured grids.

Finally, herein are some suggestions for further work to explore.

1. Expand the presented multiscale formulation to three-dimensions.
2. Study more complex physic such as compositional formulation.
3. Study non-linear flux approximations.
4. Implement and study the impact of a implicit saturation solver.
5. Implement smoothers and flux-corrective schemes.
6. Define criteria for automatic definition of coarse grids.

Bibliography

AAVATSMARK, I. et al. Discretization on unstructured grids for inhomogeneous, anisotropic media. part i: Derivation of the methods. *SIAM Journal on Scientific Computing*, v. 19, n. 5, p. 1700–1716, 1998. Cited 4 times on pages 17, 27, 28, and 29.

ARBOGAST, T.; BRYANT, S. L. A two-scale numerical subgrid technique for waterflood simulations. *SPE Journal*, v. 7, n. 04, p. 446–457, 2002. Cited on page 16.

BARBOSA, L. M. C. *Investigação de Formulações Multi-Escala Localmente Conservativas para a Simulação de Reservatórios de Petróleo Muito Heterogêneos*. Thesis (PhD) — Federal University Of Pernambuco, Recife, Brazil, 2017. Cited 10 times on pages 16, 22, 41, 49, 58, 59, 70, 73, 77, and 96.

CARVALHO, D. K. E. *Uma Formulação do Método dos Volumes Finitos com Estrutura de Dados por Aresta para a Simulação de Escoamento em Meio Poroso*. Thesis (PhD) — Federal University Of Pernambuco, Recife, Brazil, 2005. Cited 5 times on pages 20, 21, 22, 24, and 77.

CHEN, Q.-Y. et al. Enriched multi-point flux approximation for general grids. *Journal of Computational Physics*, v. 227, n. 3, p. 1701–1721, 2008. Cited 2 times on pages 17 and 29.

CHEN, Z.; HOU, T. Y. A mixed multiscale finite element method for elliptic problems with oscillating coefficients. *Mathematics of Computation*, v. 72, n. 242, p. 541–577, 2002. Cited 2 times on pages 16 and 43.

CHEN, Z.; HUAN, G.; MA, Y. *Computational Methods for Multiphase Flows in Porous Media*. 1. ed. [S.l.]: Society for Industrial Applied Math, 2006. Cited 2 times on pages 20 and 21.

CHUEH, C. et al. Multi-level adaptive simulation of transient two-phase flow in heterogeneous porous media. *Computers & Fluids*, v. 39, n. 9, p. 1585–1596, 2010. Cited on page 87.

CONTRERAS, F. et al. A cell-centered multipoint flux approximation method with a diamond stencil coupled with a higher order finite volume method for the simulation of oil-water displacements in heterogeneous and anisotropic petroleum reservoirs. *Computers & Fluids*, v. 127, p. 1–16, 2016. Cited 5 times on pages 24, 27, 29, 33, and 37.

CONTRERAS, F. R. L. *Um Método de Volumes Finitos Centrado na Célula para a Simulação de Escoamentos Bifásicos em Reservatórios de Petróleo Heterogêneos e Anisotrópicos*. Dissertação (Mestrado) — Universidade Federal de Pernambuco, 2012. Cited 7 times on pages 22, 27, 29, 30, 37, 80, and 100.

CRUMPTON, P.; SHAW, G.; WARE, A. Discretisation and multigrid solution of elliptic equations with mixed derivative terms and strongly discontinuous coefficients. *Journal of Computational Physics*, v. 116, n. 2, p. 343–358, 1995. Cited 3 times on pages 17, 27, and 29.

DURLOFSKY, L. J. Upscaling and gridding of fine scale geological models for flow simulation. *8th International Forum on Reservoir Simulation Iles Borromees*, 2005. Cited on page 16.

EDWARDS, M. G.; ROGERS, C. F. Finite volume discretization with imposed flux continuity for the general tensor pressure equation. *Computational Geosciences*, v. 2, n. 4, p. 259–290, 1998. Cited 3 times on pages 17, 27, and 29.

EDWARDS, M. G.; ZHENG, H. A quasi-positive family of continuous darcy-flux finite-volume schemes with full pressure support. *Journal of Computational Physics*, v. 227, n. 22, p. 9333–9364, 2008. Cited 3 times on pages 17, 27, and 29.

ERWING, R. Mathematical modeling and simulation for applications of fluid flow in porous media. *SIAM*, v. 1, 1983. Cited on page 19.

FARMER, C. L. Upscaling: a review. *International Journal for Numerical Methods in Fluids*, v. 40, n. 1-2, p. 63–78, 2002. Cited 2 times on pages 16 and 96.

FRIIS, H. A.; EVJE, S. Numerical treatment of two-phase flow in capillary heterogeneous porous media by finite-volume approximations. *Int. J. Numer. Anal. Mod*, v. 9, p. 505–208, 2012. Cited on page 26.

GAO, Z.; WU, J. A linearity-preserving cell-centered scheme for the heterogeneous and anisotropic diffusion equations on general meshes. *International Journal for Numerical Methods in Fluids*, v. 67, n. 12, p. 2157–2183, 2010. Cited 7 times on pages 9, 27, 29, 33, 35, 37, and 67.

GEUZAINE, C.; REMACLE, J.-F. Gmsh: A 3-d finite element mesh generator with built-in pre- and post-processing facilities. *International Journal for Numerical Methods in Engineering*, v. 79, n. 11, p. 1309–1331, 2009. Cited on page 53.

HAJIBEYGI, H.; JENNY, P. Adaptive iterative multiscale finite volume method. *Journal of Computational Physics*, v. 230, n. 3, p. 628–643, 2011. Cited 2 times on pages 49 and 70.

HELMIG, R. *Multiphase Flow and Transport Processes in the Subsurface: A Contribution to the Modeling of Hydrosystems*. 1. ed. [S.l.]: Springer-Verlag Berlin Heidelberg, 1997. Cited on page 19.

HOU, T. Y.; WU, X.-H. A multiscale finite element method for elliptic problems in composite materials and porous media. *Journal of Computational Physics*, v. 134, n. 1, p. 169–189, 1997. Cited 3 times on pages 16, 45, and 96.

HURTADO, F. S. V. *Formulação Tridimensional de Volumes Finitos para Simulação de Reservatórios de Petróleo com Malhas Não-Estruturadas Híbridas*. Thesis (PhD) — Federal University Of Santa Catarina, 2011. Cited 2 times on pages 21 and 98.

JENNY, P.; LEE, S.; TCHELEPI, H. Multi-scale finite-volume method for elliptic problems in subsurface flow simulation. *Journal of Computational Physics*, v. 187, n. 1, p. 47–67, 2003. Cited 5 times on pages 16, 39, 45, 47, and 49.

JENNY, P.; LEE, S.; TCHELEPI, H. Adaptive fully implicit multi-scale finite-volume method for multi-phase flow and transport in heterogeneous porous media. *Journal of Computational Physics*, v. 217, n. 2, p. 627–641, 2006. Cited 12 times on pages 16, 17, 39, 43, 46, 47, 48, 49, 62, 77, 96, and 101.

KARYPIS, G.; KUMAR, V. *MeTis: Unstructured Graph Partitioning and Sparse Matrix Ordering System, Version 4.0*. 2009. <http://www.cs.umn.edu/~metis>. Cited on page 53.

KIPPE, V.; AARNES, J. E.; LIE, K.-A. A comparison of multiscale methods for elliptic problems in porous media flow. *Computational Geosciences*, v. 12, n. 3, p. 377–398, 2008. Cited on page 96.

KROGSTAD, S. et al. A multiscale mixed finite-element solver for three-phase black-oil flow. *Society of Petroleum Engineers*, 2009. Cited 2 times on pages 17 and 97.

LIPNIKOV, K. et al. Monotone finite volume schemes for diffusion equations on unstructured triangular and shape-regular polygonal meshes. *Journal of Computational Physics*, v. 227, n. 1, p. 492–512, 2007. Cited on page 29.

MATLAB. *(R2014b)*. Natick, Massachusetts: The MathWorks Inc., 2014. Cited on page 67.

MOYNER, O.; LIE, K.-A. The multiscale finite volume method on unstructured grids. *SPE Reservoir Simulation Symposium*, 2013. Cited on page 17.

MOYNER, O.; LIE, K.-A. A multiscale restriction-smoothed basis method for high contrast porous media represented on unstructured grids. *Journal of Computational Physics*, v. 304, p. 46–71, 2015. Cited 10 times on pages 17, 39, 41, 43, 49, 59, 63, 73, 97, and 102.

PARRAMORE, E. et al. Multiscale finite-volume cvd-mpfa formulations on structured and unstructured grids. *Multiscale Modeling & Simulation*, v. 14, n. 2, p. 559–594, 2016. Cited on page 97.

SHENG, Z.; YUAN, G. The finite volume scheme preserving extremum principle for diffusion equations on polygonal meshes. *Journal of Computational Physics*, v. 230, n. 7, p. 2588–2604, 2011. Cited on page 29.

SOUZA, M. R. de A. *Simulação Numérica de Escoamento Bifásico em Reservatórios de Petróleo heterogêneos e Anisotrópicos Utilizando um Método de Volumes Finitos "Verdadeiramente"Multidiensional com Aproximação de Alta Ordem.* Thesis (PhD) — Federal University of Pernambuco, 2015. Cited 5 times on pages 24, 26, 27, 29, and 38.

WEISZFELD, E. Sur le point lequel la somme des distances de n points donnds est minimum. *Tohoku Mathematical*, v. 43, p. 355–386, 1937. Cited on page 58.

YUAN, G.; SHENG, Z. Monotone finite volume schemes for diffusion equations on polygonal meshes. *Journal of Computational Physics*, v. 227, n. 12, p. 6288–6312, 2008. Cited on page 80.

ZHOU, H. *Algebraic Multiscale Finite-Volume Methods for Reservoir Simulation*. Thesis (PhD) — Stanford University, 2010. Cited 7 times on pages 16, 17, 39, 43, 46, 49, and 96.

ZHOU, H.; TCHELEPI, H. A. Operator-based multiscale method for compressible flow. *SPE Journal*, v. 13, n. 02, p. 267–273, 2008. Cited 6 times on pages 16, 17, 39, 47, 96, and 101.

APPENDIX A – Resumo Estendido

Um método MsCV utilizando uma aproximação não ortodoxa MPFA-D para a simulação de escoamento bifásico em malhas verdadeiramente não estruturadas.

Introdução

Os avanços na modelagem e caracterização geoestatística permitem que informações de diferentes escalas sejam integradas para gerar modelos geocelulares cuja resolução geralmente varia de 10^8 a 10^9 células, enquanto os modelos padrão de simulação de fluxo em meio poroso podem lidar apenas com 10^6 a 10^7 blocos (ZHOU, 2010). Desta forma, múltiplas simulações diretamente em malhas de alta resolução tornam-se inviáveis. Para superar essa limitação, os métodos de transferência de escala foram planejados. Essencialmente, eles permitem que dados geoestatísticos de alta resolução sejam integrados nas malhas onde acontece a simulação de fluxo (BARBOSA, 2017).

Entre eles, destacam-se duas famílias de métodos: o *upscaling* e os métodos multiescala. O primeiro geralmente emprega um tipo de homogeneização (Farmer, 2002), mesmo quando não existe a separação formal entre as escalas. Nestes esquemas, uma solução é encontrada no espaço da escala de baixa resolução levando a resultados com baixo custo computacional e robustos, que muitas vezes não conseguem preservar detalhes do escoamento causando deterioração da precisão da representação dos fenômenos físicos estudados.

Por outro lado, a família de métodos multiescala desenvolve um conjunto de operadores, que são capazes de projetar o sistema de equações discretizado da escala fina no espaço de solução da malha menos refinada. Este sistema projetado é resolvido para então ser projetado novamente na malha de alta resolução utilizando os operadores multiescala (Hou and Wu, 1997; Jenny, Lee and Tchelepi, 2006; Zhou and Tchelepi, 2008). Isso preserva o acoplamento natural entre escalas, evitando inconsistências e perda de informações da escala fina inerentes à maioria dos métodos de upscaling.

A principal diferença entre os métodos Multiescala diz respeito à estratégia usada para impor a conservação de massa (Kippe, Aarnes and Lie, 2008). Enquanto os métodos na família do Método de Elemento Finito Misto Multiscala (MsMFEM) (Hou and Wu, 1997; Kippe, Aarnes and Lie, 2008) impõem matematicamente a conservação do fluxo criando operadores multiescala que por definição produzem campos de velocidade

conservativos, os esquemas na família de Volume Finito Multiscala (MsFV) usam uma malha grossa auxiliar adicional para reconstruir um campo de fluxo conservativo. Embora o uso de outra malha aumente a quantidade de dados armazenados, elimina consideravelmente o número de graus de liberdade ao remover a necessidade de calcular um operador multiescala que integre informações simultaneamente de campos de pressão e velocidade.

No entanto, a maioria dos métodos da família MsFVM sofrem com uma deficiência grave, eles são incapazes de trabalhar em malhas não estruturadas. Isto pode ser atribuído a três fatores; o uso da aproximação de fluxo de dois pontos (TPFA), que é consistente apenas em malhas k-ortogonais, na dificuldades apresentadas em gerar as entidades geométricas, como malhas auxiliares e centros dos volumes, e o cálculo de operadores multiscala apropriados para malhas quaisquer. Neste contexto, um grande esforço foi feito no desenvolvimento de esquemas capazes de permitir a simulação em malhas não estruturadas em ambas as escalas. Krogstad et al. (2009) desenvolveu uma variante do MsMFEM capaz de trabalhar com malhas grossas cujos volumes são quase degenerados e não estruturados. Parramore et al. (2016) desenvolveu um novo tipo de condição de contorno, Neumann-D, utilizadas para calcular o operador de prolongamento e as acoplou com um *solver* tipo MPFA-O. Moyner and Lie (2015) apresentou o método multiescala das funções de base restritivamente suavizadas (MsRSB) na qual o operador de prolongamento multiscala é calculado de modo iterativo, suavizando a função indicador de cada volume grosso restrito à região de suporte correspondente, permitindo a simulação usando malhas não estruturadas na escala de baixa resolução. Nesta dissertação, substituindo a aproximação de fluxo de dois pontos (TPFA) modificamos o método multiescala das funções de base restritivamente suavizadas (MsRSB por uma aproximação de fluxo não ortodoxa com um estêncil tipo diamante (MPFA-D) para criar o *framework* Volume de Controle Muliescala (MsCV).

Objetivos Gerais

Neste contexto, objetivo geral desta dissertação é o estudo e o desenvolvimento de métodos multiescala para a simulação do fluxo multifásico em reservatório de petróleo, utilizando malhas verdadeiramente não estruturadas.

Objetivos Específicos

1. Desenvolver um estratégia multiscala para simulação de fluxo monofásico e bifásico em reservatório de petróleo altamente heterogêneo e anisotrópico usando uma aproximação de fluxo de múltiplos pontos consistente com malhas não estruturadas em qualquer escala.

2. Incorporar uma aproximação MPFA-D ao resolvedor de pressão multiescala.
3. Criar ferramentas alternativas para realizar o engrossamento da malha de alta resolução.

Formulação Matemática

Este trabalho utiliza uma formulação segregada que deriva a partir das leis de conservação de massa duas equações diferenciais parciais a serem resolvidas sequencialmente, a equação da pressão de natureza elíptica e a equação de saturação de natureza hiperbólica.

Equação da Pressão

A equação da pressão é uma equação diferencial parcial de natureza elíptica que descreve o campo de pressão no reservatório de petróleo submetido a escoamento bifásico incompressível. Para nosso problema, a equação da pressão pode ser definida como:

$$-\vec{\nabla}\cdot\vec{v} = Q \quad \text{com} \quad \vec{v} = -\lambda\underline{K}(\vec{\nabla}p - \rho_{\text{avg}}\vec{g}) \quad \text{in} \quad \Omega \tag{A.1}$$

onde $\vec{v} = \vec{v}_w + \vec{v}_o$ é a velocidade de Darcy total do fluido negligenciando a compressibilidade, $\vec{\nabla}p$ é o gradiente de pressão e $Q = Q_w + Q_o$ é a soma do termo fonte de cada fase dividdo pela densidade correspondente, $Q_i = q_i/\rho_i$. A densidade ponderada ρ_{avg} é a média das densidades ponderadas pelas mobilidades (HURTADO, 2011). Nessa trabalho os efeitos da gravidade são desconsiderados.

$$\rho_{\text{avg}} = \frac{\lambda_o\rho_o + \lambda_w\rho_w}{\lambda} \quad \text{com} \quad \lambda = \lambda_o + \lambda_w \tag{A.2}$$

Equação da Saturação

A equação de saturação é uma equação diferencial parcial hiperbólica não-linear que descreve como uma das fases é transportada através do meio poroso:

$$\phi\frac{\partial S_w}{\partial t} = -\vec{\nabla}\cdot\vec{F}(S_w) + Q_w \quad \text{para} \quad \Omega\times[0,t] \tag{A.3}$$

onde S_w é a saturação da fase água e Q_w é o termo fonte da fase água e onde $\vec{F}(S_w)$ é definido como:

$$\vec{F}(S_w) = f_w(S_w)\vec{v} \quad \text{com} \quad f_w = \lambda_w/\lambda \tag{A.4}$$

O fluxo fracional $f_w(S_w)$ expressa a fração de água do fluxo com respeito ao fluxo total. Note que o fluxo fracional depende da saturação, o que torna esta equação

não-linear. Por fim, a suposição que o meio encontra-se totalmente saturado nos dá a equação de fechamento:

$$S_o + S_w = 1 \tag{A.5}$$

Condições Iniciais e de Contorno

Para tornar o modelo matemático descrito acima bem-posto, devemos propor condições de contorno e condições iniciais consistentes. Além disso, devemos considerar as condições que representem a iteração dos poços com o meio poroso. A injeção em poços a taxa de fluxo q_i é equivalente a uma condição de Neumann, enquanto a produção a pressão controlada q_p no poço de extração é considerada uma condição de Dirichlet.

As condições de contorno mais usuais para equação da pressão são:

$$\begin{aligned} &p(\vec{x},t) = g_D \quad \text{em} \quad \partial\Omega_D \\ &\vec{v}\cdot\vec{n} = g_N \quad \text{em} \quad \partial\Omega_N \\ &p(\vec{x},t) = p_{\text{produção}} \quad \text{em} \quad \partial\Omega_p \\ &\vec{v}\cdot\vec{n} = Q_{\text{injeção}} \quad \text{em} \quad \partial\Omega_i \end{aligned} \tag{A.6}$$

A primeiras duas condições na equação acima são respectivamente condições de Dirichlet (g_D) e Neumann (g_N), isto é condição de pressão ou fluxo prescrito nos contornos do domínio. As demais condições são também condições de Dirichlet e Neumman mas definidas nos poços produtores ($p_{\text{produção}}$) e injetores $Q_{\text{injeção}}$.

Já a equação da saturação precisa de uma definição inicial do campo de saturação e da definição da saturação de água sendo atravessando poço injetor.

$$\begin{aligned} &S_w(\vec{x},t) = \overline{S}_w \quad \text{em} \quad \partial\Omega \times [0,t] \\ &S_w(\vec{x},0) = \overline{S}_w^0 \quad \text{em} \quad \Omega \end{aligned} \tag{A.7}$$

Formulação Numérica

As equações da pressão e saturação são então discretizadas utilizando esquemas na família dos volumes finitos. O sistema de equações resultante é reacoplado através de uma variante da metodologia IMPES (pressão implícita e saturação explicita) onde o *solver* de pressão é substituído por um *solver* de multiescala para pressão e vazão.

Equação da Pressão Discreta

Utilizamos um método não ortodoxo de aproximação de fluxo onde o gradiente de pressão é aproximado em uma aresta é aproximado com múltiplos pontos em um

estêncil que se aproxima a de um diamante.

$$\vec{v}_{IJ} \cdot \vec{N}_{IJ} \simeq \tau_{IJ}[p_{\hat{R}} - p_{\hat{L}} - \upsilon_{IJ}(p_J - p_I)] \tag{A.8}$$

onde a transmissibilidade escalar τ_{IJ}, e o parâmetro tangencial não-dimensional υ_{IJ} são dados por:

$$\tau_{IJ} = -\lambda_{IJ}|IJ| \frac{K^n_{IJ(\hat{L})} K^n_{IJ(\hat{R})}}{K^n_{IJ(\hat{L})} h^{\hat{R}}_{IJ} + K^n_{IJ(\hat{R})} h^{\hat{L}}_{IJ}} \tag{A.9}$$

$$\upsilon_{IJ} = \frac{\overrightarrow{IJ} \cdot \overrightarrow{\hat{L}\hat{R}}}{|IJ|^2} - \frac{1}{|IJ|}\left(\frac{K^t_{IJ(\hat{L})}}{K^n_{IJ(\hat{L})}} h^{\hat{L}}_{IJ} + \frac{K^t_{IJ(\hat{R})}}{K^n_{IJ(\hat{R})}} h^{\hat{R}}_{IJ} \right) \tag{A.10}$$

Tratamento das Condições de Contorno

Pode-se mostrar que as arestas submetidas a condição de Dirichlet tem o seu fluxo definidos por (CONTRERAS, 2012):

$$\vec{v}_{IJ} \cdot \vec{N}_{IJ} \simeq -\frac{\lambda_{IJ} K^n_{IJ}}{h^{\hat{L}}_{IJ}|IJ|} \left[(\overrightarrow{J\hat{L}} \cdot \overrightarrow{JI}) g_D(I) + \overrightarrow{I\hat{L}} \cdot \overrightarrow{IJ}) g_D(J) - p_{\hat{L}} |IL|^2 \right] - K^t_{IJ} \left(g_D(J) - g_D(I) \right) \tag{A.11}$$

onde $g_D(I)$ e $g_D(J)$ são as pressões de Dirichlet.

Já as arestas submetidas as condição de Neumann tem fluxo definido como:

$$\vec{v}_{IJ} \cdot \vec{N}_{IJ} = g_N |IJ| \tag{A.12}$$

onde g_N é a velocidade normal a aresta IJ.

Interpolação Conservativa

Para que a formulação acima continue centrada na célula e coerente com os métodos dos volumes finitos, deve-se utilizar algum tipo de interpolação conservativas para reescrever as pressões p_i e p_j em função da pressão dos elementos ao seu redor. Essa interpolação é apresentada pela Equação 3.46.

Equação da Saturação Discreta

Utilizando o método de Euler para discretização temporal e uma aproximação de primeira ordem para o termo espacial obtemos a equação da saturação discretizada:

$$S^{n+1}_{w,\hat{k}} = S^n_{w,\hat{k}} - \frac{\Delta t}{\phi_{\hat{k}} V_{\hat{k}}} \sum_{IJ \in \Omega_{\hat{k}}} \vec{F}(S_w)_{IJ} \cdot \vec{N}_{IJ} + \frac{\Delta t \overline{Q}_w}{\phi_{\hat{k}} V_{\hat{k}}} \tag{A.13}$$

onde a saturação do passo posterior $S_{w,\hat{k}}^{n+1}$ é aproximada em função da saturação do passo atual $S_{w,\hat{k}}^{n}$, da porosidade $\phi_{\hat{k}}$, do volume do elemento $V_{\hat{k}}$ e onde o termo hiperbólico é $\vec{F}(S_w)_{IJ} = f_w\left(S_{w,\hat{k}}^{n}\right)\vec{v}_{IJ}$ e o passo de tempo é $\Delta t = t^{n+1} - t^n$.

Formulação Multiescala

A definição de métodos multiscala surge da necessidade de troca de informações em diferentes escalas. Como simulação direta em uma malha de alta resolução (malha fina Ω) não é viável, o método multiescala projetas o sistema de equações de alta resolução no espaço da malha de menor resolução (malha grossa primal, Ω_c) usando os operadores multiescala. O sistema de equações, agora com baixa resolução, é resolvido e sua solução é projetada de na malha de alta resolução usando o operador de prolongamento. Assim, o sistema de resolução inferior funciona como base auxiliar com um número considerável menor de graus de liberdade criado para permitir a simulação, enquanto os operadores de multiescala capturam o acoplamento entre essas escalas. Zhou and Tchelepi (2008) desenvolveram uma notação algébrica baseada no trabalho de Jenny, Lee and Tchelepi (2006) que descreve esses operadores de transferência de escala em notação matricial com o operador de Restrição e Prolongamento respectivamente, definido como:

$$Q_c = \underset{\sim}{R}_{op} Q_f \tag{A.14}$$

$$P_f = \underset{\sim}{P}_{op} P_c \tag{A.15}$$

onde P_f é um vetor solução $n_f \times 1$ da pressão na escala fina, P_c é um vetor solução $n_c \times 1$ da escala grossa, $\underset{\sim}{P}_{op}$ uma matriz $n_f \times n_c$ e $\underset{\sim}{R}_{op}$ uma matriz $n_c \times n_f$ e Q_f o vetor que armazena termos fonte/sumidou na escala fina e Q_c o mesmo vetor mapeado na escala de baixa resolução.

Após manipulações algébricas, definimos o sistema de equações na escala de baixa resolução como:

$$\underset{\sim}{T}_c P_c = Q_c \tag{A.16}$$

Onde $\underset{\sim}{T}_c$ é a matriz de transmissibilidade na escala grossa é definida por:

$$\underset{\sim}{T}_c = \underset{\sim}{R}_{op} \underset{\sim}{T}_f \underset{\sim}{P}_{op} \tag{A.17}$$

sabendo que $\underset{\sim}{T}_f$ é a matriz de transmissibilidade na escala fina.

Métodos Multiescala das Funções de Base Restritamente Suavizadas (MsRSB)

O MsRSB desenvolvido por Moyner and Lie (2015) tem como sua principal vantagem a possibilidade de trabalhar com malhas não estruturadas na escala grossa. Essa flexibilidade permite a construção de malhas na grossas primais de baixa resolução que se adaptem à meios com falhas, com canais de alta e baixa permeabilidade. O *framework* do MsRSB herda estruturas e definições geométricas básicas do MsFV, como malha fina, malha grossa primal, centro do volume da malha grossa primal, região de suporte de um centro. Entretanto, os algoritmos de pré-processamento são generalizações dos algoritmos clássicos para trabalhar em malhas na escala grossa não estruturadas. Além disso o MsRSB, assim como a maioria dos métodos na família MsFV, emprega TPFA que é apenas consistente em malhas k-ortogonais. Isto limita severamente a sua aplicação em geometrias complexas que restringem o seu uso a malhas finas estruturadas. O MsCV modifica substitui a aproximação tipo TPFA por uma aproximação tipo MPFA-D. Abaixo segue a lista das entidades geométricas necessárias para método MsRSB.

Malha da Escala Fina (Ω^f or Ω): Esta é a malha de alta resolução oriunda da discretização do domínio físico. Métodos multiescala tradicionais geralmente utilizam algum tipo de malha k-ortogonal.

Malha Grossa Primal (Ω_c^p ou Ω^p): É a malha de baixa resolução gerada pelo aglomeramento de elementos da malha da escala fina.

Centro de um Volume da Malha Grossa Primal (x^P): É um volume da malha fina que representa o centro de um volume da malha grossa primal. Metodos clássicos utilizam o volume fino mais próximo ao centroide.

Região de Supote de um Volume da Malha Primal j (I_j): A definição de região de suporte de um volume primal serve ao mesmo propósito da definição da malha dual do MsFV, calcular o operador de prolongamento e reimpor conservação de massa na fronteira dos elementos da malha. Além disso a definição matemática delimita todos os elementos da malha fina tais que

$$(\underline{P}_{op})_{i,j} > 0 \quad \forall i \in I_j, \quad \text{caso contrário} \quad (\underline{P}_{op})_{i,j} = 0$$

I_j define a zona de influência do centro de um volume da malha grossa primal x_j^P. O MsRSB usa os volumes da malha grossa primal e seus centros para criar a região de suporte de um volume grosso primal para definir todas as outras entidades geométricas.

Fronteiras da Região de Suporte, (B_j)**:** A fronteira da região de suporte consiste em todos os volumes da malha fina que compartilham ao menos uma aresta com I_j mas não estão contidos nela.

Fronteira Global das Regiões de Suporte, (G)**:** A região de fronteira global das regiões de suporte é a união de todas as fronteiras da região de suporte de cada volume da malha grossa primal.

Fronteira Global das Regiões de Suporte em uma Região de Suporte, H_j**:** Consiste na interseção de I_j com G.

Operador de Restrição

O operador de Restrição do MsRSB e do MsCV é definido como:

$$(\underset{\sim}{R}_{op})_{ij} = \begin{cases} 1 & \Omega_i^f \in \Omega_j^c; \\ 0, & \text{caso contrário} \end{cases} \quad \text{onde} \quad 1 \leqslant j \leqslant n_c \quad \text{e} \quad 1 \leqslant i \leqslant n_f \tag{A.18}$$

onde n_f e n_c são o número de volumes na malha fina e na malha grossa primal respectivamente.

Operador de Prolongamento

O operador de prolongamento é calculado resolvendo a parte homogênea da equação da pressão. Condições de contorno especificas são utilizadas para desacoplar o problema gerando múltiplos problemas pequenos. Como consequência este operador armazena em cada coluna a solução da equação da pressão normalizada resolvida na região de suporte correspondente. O MsRSB e o MsCV utilizam uma versão modificada do método de Jacobi ponderado para definição do operador de prolongamento:

1. Calcula-se o incremento do método de Jacobi ponderado:

$$\widehat{d_j} = -\omega D^{-1} \underset{\sim}{T}_f^{pre} (\underset{\sim}{P}_{op})_j^{\eta} \tag{A.19}$$

onde ω é o parametro de amortecimento do método de Jacobi ponderado, D é a matriz diagonal que contém a diagonal da matriz de transmissibilidade pré-condicionada $\underset{\sim}{T}_f^{pre}$ e definida por:

$$(\underset{\sim}{T}_f^{pre})_{ij} = \begin{cases} (\mathcal{T}_f)_{ij} & i \neq j; \\ (\mathcal{T}_f)_{ij} - \sum_{k=1}^{n} (\mathcal{T}_f)_{ik} & i = j; \end{cases} \tag{A.20}$$

2. O incremento do Método de Jacobi ponderado é modificado de acordo com a seguinte função. Perceba que esta é uma função por partes e que restringe o crescimento fora da região de suporte sem perder a partição da unidade.

$$d_{ij} = \begin{cases} \frac{\widehat{d}_{ij} - (\underset{\sim}{P}_{op})_{ij}^{\eta} \sum_{k \in H_k} \widehat{d}_{ik}}{1 + \sum_{k \in H_k} \widehat{d}_{ik}} & \Omega_j^f \in H_j; \\ \widehat{d}_{ij} & \Omega_j^f \in I_j \text{ e } \Omega_j^f \notin H_j; \\ 0 & \Omega_j^f \notin I_j; \end{cases} \tag{A.21}$$

3. O operador de prolongamento é atualizado utilizando o incremento modificado.

$$(\underset{\sim}{P}_{op})_j^{\eta+1} = (\underset{\sim}{P}_{op})_j^{\eta} + d_j \tag{A.22}$$

4. Calcula-se o erro nos volumes finos não pertencentes a Fronteira Global das Regiões de Suporte.

$$e_j = \max_i(\|\widehat{d}_{ij}\|), \quad i \notin G \tag{A.23}$$

5. Testa convergência. Se $\|e\|_\infty > tol$ o atual $(\underset{\sim}{P}_{op})_j^{\eta}$ é utilizado como solução inicial do primeiro passo e este processo repetido.

Conclusões

Nesta dissertação, modificamos o método multiescala das funções de base restritivamente suavizadas (MsRSB) substituindo a aproximação de fluxo de dois pontos (TPFA) por uma aproximação de fluxo não ortodoxa com um estêncil tipo diamante (MPFA-D) para criar o *framework* Volume de Controle Muliescala (MsCV) que permite a simulação de malhas não estruturadas em ambas as escalas.

Também foi desenvolvido algoritmos para gerar malhas grossas primais adaptativas a partir de quaisquer tipo de malhas de alta resolução. Os exemplos mostraram que embora a aproximação MPFA-D seja linear por partes, o MsCV não preserva esta propriedade. Entretanto, isso não teve impacto nas soluções de escoamento bifásico.

Além disso, as rotinas desenvolvidas para geração de malhas grossa primal em quase todos os casos obtiveram resultados melhores que as malhas geradas no Metis. Este conjunto de rotinas que permitem ao usuário gerar malhas adaptáveis as propriedades geofísicas do reservatório mostrou excelentes resultados. Em geral, o MsCV foi capaz de obter bons resultados na simulação de escoamento água-óleo em meio poroso altamente heterogêneo e levemente anisotrópico preservando as características de alta resolução do escoamento.

Ademais, a habilidade de trabalhar com malhas verdadeiramente não estruturadas em ambas as escolas permite uma discretização mais adequada das formações

geológicas complexas, permitindo a simulação multiescala em malhas finas com um número consideravelmente menor de graus de liberdade em comparação com as malhas estruturadas equivalentes.

Em conclusão, o *framework* do MsCV é capaz de produzir soluções acuradas a custos computacionais razoáveis para simulação de escoamento bifásico em meio poroso heterogêneo e anisotrópico utilizando malhas verdadeiramente não estruturadas.