



**Universidade do Estado do Rio de Janeiro**

Centro de Tecnologia e Ciências

Faculdade de Engenharia

Hyun Ho Shin

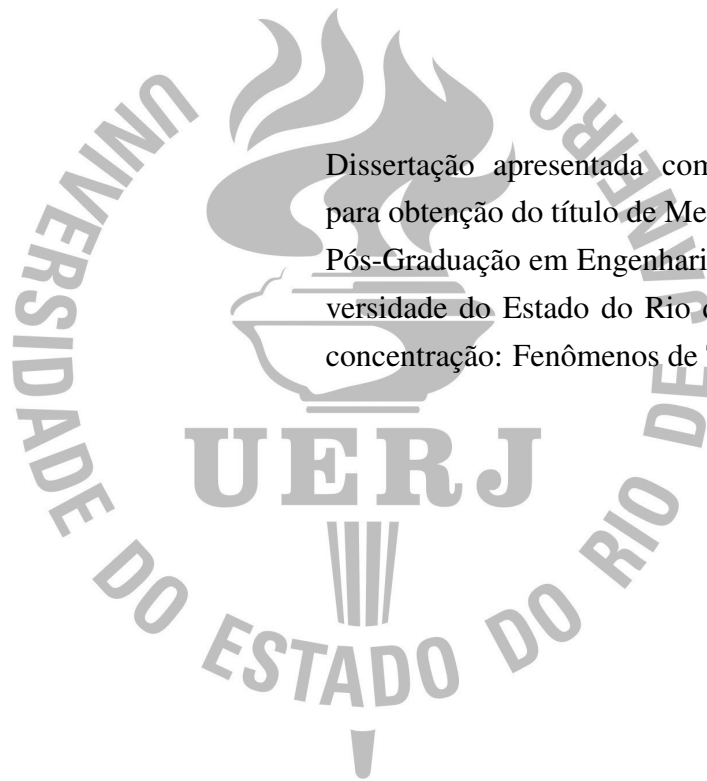
**A methodology of study of three dimensional stratified turbulent fluid flow  
for hydroelectric power plant reservoir simulation**

Rio de Janeiro

2009

Hyun Ho Shin

**A methodology of study of three dimensional stratified turbulent fluid flow  
for hydroelectric power plant reservoir simulation**



Dissertação apresentada como requisito parcial para obtenção do título de Mestre, ao Programa de Pós-Graduação em Engenharia Mecânica, da Universidade do Estado do Rio de Janeiro. Área de concentração: Fenômenos de Transporte.

Advisor: Prof. Dr. Norberto Mangiavacchi

Co-Advisor: Prof. Dr. Christian Schaerer

Rio de Janeiro

2009

CATALOGAÇÃO NA FONTE  
UERJ / REDE SIRIUS / BIBLIOTECA CTC/B

S557 Shin, Hyun Ho

A methodology of study of three dimensional stratified turbulent fluid flow for hydroelectric power plant reservoir simulation / Hyun Ho Shin. – 2009.

83f.

Orientador: Norberto Mangiavacchi.

Coorientador: Christian Schaerer.

Dissertação (Mestrado) – Universidade do Estado do Rio de Janeiro, Faculdade de Engenharia.

1. Engenharia Mecânica. 2. Hidrodinâmica – Dissertação. 3. I. Mangiavacchi, Norberto. II. Universidade do Estado do Rio de Janeiro. III. Título.

CDU 621

Autorizo, apenas para fins acadêmicos e científicos, a reprodução total ou parcial desta dissertação, desde que citada a fonte.

---

Assinatura

---

Data

Hyun Ho Shin

**A methodology of study of three dimensional stratified turbulent fluid flow for hydroelectric power plant reservoir simulation**

Dissertação apresentada como requisito parcial para obtenção do título de Mestre, ao Programa de Pós-Graduação em Engenharia Mecânica, da Universidade do Estado do Rio de Janeiro. Área de concentração: Fenômenos de Transporte.

Aprovado em 30 de Junho de 2009

Banca Examinadora:

---

Prof. Dr. Norberto Mangiavacchi (Orientador)  
Faculdade de Engenharia Mecânica – UERJ

---

Prof. Dr. Christian Emilio Schaerer Serra (Coorientador)  
Faculdade de Politécnica – UNA

---

Prof. Dr. Carlos Antônio de Moura  
Instituto de Matemática e Estatística – UERJ

---

Prof. Dr. Alvaro Luiz Gayoso de Azeredo Coutinho  
Universidade Federal do Rio de Janeiro – UFRJ

---

Prof. Dr. Edson Cezar Wendland  
Universidade de São Paulo – USP

---

Prof. Dr. José da Rocha Miranda Pontes  
Universidade Federal do Rio de Janeiro - UFRJ

---

Dr. Cássio Pereira Botelho Soares  
Furnas Centrais Elétricas S.A.

Rio de Janeiro  
2009

## ACKNOWLEDGMENTS

I wish to thank my advisor, Professor Norberto Mangiavacchi, for all his help. It was absolutely invaluable. He guided me, with endless patience, during two years of development and in the redaction of this dissertation.

Special thanks to Professor Christian Schaerer, my mentor and friend, for showing me the option to study at the UERJ (Rio de Janeiro), and for the moments we shared together.

Thanks to Cássio Botelho for the tips he gave me for this work and for the moral support.

Thanks to CAPES and FURNAS for financial support.

Dozens of people have helped and taught me at the GESAR (development group at the UERJ) and at the post-graduation of the UERJ. Sonia Nina welcomed me to the GESAR, and she has showed interest in my well-being. Maxini Matos has overcome the bureaucratic procedures to facilitate my studies. Gustavo Rabello has left the hydrodynamic simulator, in which my work is based. Thank you to those who helped the project as staff: Abrão, André, Cristiane, Eduardo, George, Hugo, Manolo, Marcia, Mariana, Pedro C., Renan, Virginia, Wagner. Thanks also to professors Manoel, Mariano, Mila, Moura, Soriano, Rogerio.

Special thanks to my friends and project partners: Pedro Torres, for the tips he gave me to solve the linear system equations and for the moral support; Leon Matos and Felipe Santiago, for their help and for mutual support during the difficult times of the project.

Many thanks to my second family, *tia* Elza and Thiago, for accepting me as part of their family in Rio de Janeiro. Those who made me feel part of their families: Marquinhos, Carlos V., Timóteo, Paolinha, Journal, Luciana, Lorna, *tia* Guga and Paolo. Thank you very much!!! You are special in my life!!

RIO-OMC (Rio de Janeiro Oriental Missionary Church) led by the Pastor Daniel Oh and his wife Helena, the young group: Hauley, Romulo, Pri, Alex, John, Leo, Alexandre, Samuel, Fernando, Esther, Sara, Mariana, and adults. Thanks for making me part of their community and for being such a beautiful part of my life.

Thanks to RED (University Christian Network in Paraguay) led by Mirna, Eduardo L., Eduardo D., Mirian and Samuel, despite the distance, for their affection for me.

Thanks to Damaris, Carlos and Luz, Laura, and Alicia, for the friendship.

Special thanks to Byung Min, my soul brother, for moments we shared together and for moral support. God has blessed me through our friendship!

Very special thanks to my dad, my mom and my sister. Thank you for teaching God's

love and for your prayers. I thank God for your lives. Thanks Shin In Soo, Ko Soon Duk and Shin Hyun Ju! You are very special! I love you!

Jesus Christ, thank you for accepting me and loving me as I am. Even when I am unfaithful You are always faithful. You never left me alone. You were always there! Thanks God for the blessings of knowing so many special people! Thank You for everything You've done and for everything You will do! Thank You my sweet Lord! You are the One and there is no One like You! And I love You!

Oh, the depth of the riches of the wisdom and knowledge of God!

How unsearchable his judgments, and his paths beyond tracing out!

*"Who has known the mind of the Lord? Or who has been his counselor?"*

*"Who has ever given to God, that God should repay him?"*

For from him and through him and to him are all things. To him be the glory forever! Amen.

*Romans 11:33–36*

## RESUMO

SHIN, Hyun Ho. *Uma Metodologia de Estudo de Simulação Tridimensional de Escoamento Turbulento Estratificado no Reservatório de Plantas Hidrelétricas*. 2009. 83f. Dissertação (Mestrado em Engenharia Mecânica) – Faculdade de Engenharia Mecânica, Universidade do Estado do Rio de Janeiro, Rio de Janeiro, 2009.

Uma simulação numérica que leva em conta os efeitos de estratificação e mistura escalar (como a temperatura, salinidade ou substância solúvel em água) é necessária para estudar e prever os impactos ambientais que um reservatório de usina hidrelétrica pode produzir. Este trabalho sugere uma metodologia para o estudo de escoamentos ambientais, principalmente aqueles em que o conhecimento da interação entre a estratificação e mistura pode dar noções importantes dos fenômenos que ocorrem. Por esta razão, ferramentas de simulação numérica 3D de escoamento ambiental são desenvolvidas. Um gerador de malha de tetraedros do reservatório e o modelo de turbulência algébrico baseado no número de Richardson são as principais ferramentas desenvolvidas. A principal dificuldade na geração de uma malha de tetraedros de um reservatório é a distribuição não uniforme dos pontos relacionada com a relação desproporcional entre as escalas horizontais e verticais do reservatório. Neste tipo de distribuição de pontos, o algoritmo convencional de geração de malha de tetraedros pode tornar-se instável. Por esta razão, um gerador de malha não estruturada de tetraedros é desenvolvido e a metodologia utilizada para obter elementos conformes é descrita. A geração de malha superficial de triângulos utilizando a triangulação Delaunay e a construção do tetraedros a partir da malha triangular são os principais passos para o gerador de malha. A simulação hidrodinâmica com o modelo de turbulência fornece uma ferramenta útil e computacionalmente viável para fins de engenharia. Além disso, o modelo de turbulência baseado no número de Richardson leva em conta os efeitos da interação entre turbulência e estratificação. O modelo algébrico é o mais simples entre os diversos modelos de turbulência. Mas, fornece resultados realistas com o ajuste de uma pequena quantidade de parâmetros. São incorporados os modelos de viscosidade/difusividade turbulenta para escoamento estratificado. Na aproximação das equações médias de Reynolds e transporte de escalar é utilizando o Método dos Elementos Finitos. Os termos convectivos são aproximados utilizando o método semi-Lagrangeano, e a aproximação espacial é baseada no método de Galerkin. Os resultados computacionais são comparados com os resultados disponíveis na literatura. E, finalmente, a simulação de escoamento em um braço de reservatório é apresentada.

Palavras-chave: Geração de Malha não estruturada, Simulação Numérica, Métodos de Elementos Finitos, Equações Médias de Reynolds, Modelo de Viscosidade/Difusividade Turbulenta, Escoamento Estratificado Turbulento.



## ABSTRACT

To study and forecast the environmental impacts that a hydroelectric power plant reservoir may produce, a numerical simulation that takes into account the effects of stratification and scalar mixing (such as temperature, salinity or water-soluble substance) is required. This work proposes a methodology for the study of the environmental fluid flow phenomena, mainly for flows in which the knowledge of the interaction between stratification and mixing can give important notions of the phenomena that occur. For this, a numerical simulation tool for 3D environmental flow is developed. A tetrahedral mesh generator of the reservoir based on the terrain topology and an algebraic turbulence model based on the Richardson number are the main tools developed. The main difficulty in tetrahedral mesh generation of a reservoir is non-uniform distribution of the points related to the huge ratio between the horizontal and vertical scales of the reservoir. In this type of point distributions, conventional tetrahedron mesh generation algorithm may become unstable. For this reason, a unstructured tetrahedral mesh generator is developed and the methodology used to obtain conforming elements is described. Triangular surface mesh generation using the Delaunay triangulation and the construction of the tetrahedra from the triangular surface mesh are the main steps to the mesh generator. The hydrodynamic simulation of reservoirs with a turbulence model provides a useful tool that is computationally viable for engineering purposes. Furthermore, the turbulence model based on the Richardson number takes into account the effects of interaction between turbulence and stratification. The algebraic model is the simplest among the various models of turbulence, but provides realistic results with the fitting of a small amount of parameters. Eddy-Viscosity/Diffusivity models for stratified turbulent flows models are incorporated. Using the Finite Element Method (FEM) approximation the Reynolds-averaged Navier-Stokes (RANS) and mean scalar transport equations are approximated. The convective terms are discretized employing the Semi-Lagrangian method, and the spatial discretization is based on the Galerkin method. The computational results are compared with the results available in the literature. Finally, the simulation of the flow in a branch of a reservoir is presented.

**Keywords:** Unstructured Mesh Generation, Numerical Simulation, Finite Element Method, Reynolds Averaged Navier Stokes equations, Eddy-Viscosity/Diffusivity Models, Stratified Turbulent Flow.

---

## LIST OF FIGURES

---

1.1	Images of the raster data: (a) GeoTIFF image of a region in <i>Rio de Janeiro</i> state ( $43^{\circ}$ S – $22^{\circ}$ W). (b) DTM image of the region of <i>Rio de Janeiro</i> state and south of <i>Minas Gerais</i> state. . . . .	18
1.2	Images of the vector data: (a) Shapefile of a region of <i>Nova Friburgo</i> in <i>Rio de Janeiro</i> state. (b) Some contour lines of a river channel extrated from the shapefile showed in (a). . . . .	19
1.3	Image of the contour lines with the cloud of points. . . . .	20
2.1	Non-uniform distribution of points in a river channel: (a) Side view. (b) Top view. . . . .	21
2.2	Randomized incremental algorithm of node insertion: (a) Given an initial triangulation, triangles with circumcircles containing a new point are removed. (b) A new triangulation is shown. (MAVRIP LIS, 1997) . . . . .	22
2.3	Points adding process: (a) contour lines points. (b) points filling the reservoir. . . . .	26
2.4	Partitioning prisms: (a) non-conforming tetrahedron. (b) conforming tetrahedron. . . . .	27
2.5	Three-dimensional Mesh. . . . .	27
3.1	Prandtl's mixing length: average velocity profile in the boundary layer of a two-dimensional flow. . . . .	37
3.2	Mixing length model for turbulent boundary layer. . . . .	40

---

4.1	Mini tetrahedron element. . . . .	49
5.1	UML class diagram of the <i>GUI</i> . . . . .	61
5.2	UML class diagram of the <i>Simulator</i> . . . . .	62
6.1	Initial configuration of the 2D gavity current simulation. . . . .	65
6.2	Comparison between the experimental and numerical results of Eghbalzadeh et al. (2008) and the simulation performed: (a) $t = 4.4 s$ . (b) $t = 6.8 s$ . . . . .	66
6.3	Velocity of the computed gravity current nose. . . . .	66
6.4	Scale model of confluence: (a) Sketch. (b) Picture. . . . .	68
6.5	Mesh of the confluence model: (a) Surface mesh. (b) 3D mesh with edges of the tetrahedra. . . . .	68
6.6	Numerical simulation of gravity current wave propagating along the tributary. . . . .	69
6.7	Propagation of the wave similar to an axisymmetric gravity current: (a) Laboratory experiment. (b) Numerical simulation. . . . .	69
6.8	Propagation of the wave in the main stem: (a) Laboratory experiment. (b) Numerical simulation. . . . .	70
6.9	Propagation of the wave in the main stem: (a) Laboratory experiment. (b) Numerical simulation. . . . .	70
6.10	Contour lines and the dimensions of the reservoir <i>GUI</i> . . . . .	71
6.11	Contour lines and cloud of point shown in the interface <i>GUI</i> . . . . .	71
6.12	2D triangular mesh shown in the <i>GUI</i> . . . . .	72
6.13	Mesh of the reservoir: (a) Surface mesh (top view). (b) 3D mesh. . . . .	72
6.14	Initial condition of the simulation. . . . .	73
6.15	Simulation at 14.87 s. . . . .	73
6.16	Simulation at 29.74 s. . . . .	74
6.17	Simulation at 59.48 s. . . . .	74
6.18	Simulation at 89.22 s. . . . .	75
6.19	Simulation at 178.44 s. . . . .	75

---

# CONTENTS

---

INTRODUCTION . . . . .	13
1 GEOGRAPHIC TOPOLOGY . . . . .	17
<b>1.1 Raster and Vector data type</b> . . . . .	17
1.1.1 Raster data type . . . . .	17
1.1.2 Vector data type . . . . .	19
2 MESH GENERATION . . . . .	21
<b>2.1 Delaunay triangulation</b> . . . . .	22
2.1.1 Randomized incremental algorithm . . . . .	22
2.1.2 Convex hull . . . . .	23
2.1.3 Non-uniform distribution of points . . . . .	24
2.1.4 Optimized search . . . . .	25
<b>2.2 Reservoir 3D Mesh Generation Routine</b> . . . . .	25
2.2.1 Points adding process . . . . .	26
2.2.2 Prism partition process . . . . .	26
3 GOVERNING EQUATIONS . . . . .	28
<b>3.1 Fundamental equations of fluid flow</b> . . . . .	28
3.1.1 Continuity equation . . . . .	28

---

3.1.2	Navier-Stokes equation . . . . .	29
3.1.3	Scalar transport equation . . . . .	31
<b>3.2</b>	<b>Derivation of mean equations . . . . .</b>	<b>31</b>
3.2.1	Statistical approach . . . . .	31
3.2.2	Mean continuity equation . . . . .	32
3.2.3	Reynolds averaged Navier-Stokes equations . . . . .	33
3.2.4	Mean scalar transport equation . . . . .	34
3.2.5	The closure problem . . . . .	34
<b>3.3</b>	<b>Eddy-viscosity/diffusivity model . . . . .</b>	<b>35</b>
<b>3.4</b>	<b>Algebraic models . . . . .</b>	<b>36</b>
<b>3.5</b>	<b>Boundary layer . . . . .</b>	<b>38</b>
<b>3.6</b>	<b>Turbulence modeling for stratified flows . . . . .</b>	<b>40</b>
<b>3.7</b>	<b>Boundary and initial condition . . . . .</b>	<b>41</b>
<b>3.8</b>	<b>Nondimensional form . . . . .</b>	<b>42</b>
<b>4</b>	<b>DISCRETIZATION METHOD . . . . .</b>	<b>45</b>
<b>4.1</b>	<b>Variational approaches . . . . .</b>	<b>45</b>
<b>4.2</b>	<b>Element of mesh . . . . .</b>	<b>48</b>
4.2.1	Volume coordinates . . . . .	49
4.2.2	Shape function for tetrahedra . . . . .	50
<b>4.3</b>	<b>Galerkin method . . . . .</b>	<b>51</b>
<b>4.4</b>	<b>Semi-Lagrangian method . . . . .</b>	<b>56</b>
<b>4.5</b>	<b>Projection method . . . . .</b>	<b>57</b>
<b>5</b>	<b>SYSTEM DESIGN . . . . .</b>	<b>60</b>
<b>6</b>	<b>NUMERICAL SIMULATIONS . . . . .</b>	<b>64</b>
<b>6.1</b>	<b>2D Gravity Current . . . . .</b>	<b>65</b>
<b>6.2</b>	<b>Model of confluence . . . . .</b>	<b>67</b>
6.2.1	Laboratory Experiment . . . . .	67
6.2.2	Numerical Simulation . . . . .	67
6.2.3	Results . . . . .	69

CONTENTS	12
<b>6.3 Reservoir Simulation</b> . . . . .	70
7 SUMMARY . . . . .	77
<b>7.1 Future Research Area</b> . . . . .	79
REFERENCES . . . . .	80

---

## INTRODUCTION

---

Vertical profiles of temperature in the water column of several reservoirs show the stratified condition of the reservoirs (GUÉRIN et al., 2006; DEMARTY; BASTIEN; TREMBLAY, 2009). This condition affects the distribution of the nutrients amount and also the contaminants may exist. In temperate regions, the climate in summer and winter seasons are favorable to the thermal stratification. However, in spring and autumn, the thermal stratification is broken due to the overturn produced by the variation in temperature profile, that leads to mixing of the water column and the nutrients. The result of this mixing is water with homogenized properties. In tropical regions, the reservoir is stratified most of the year. But in wet season, rain and wind provide the necessary energy for mixing.

The knowledge of the effects of stratification and the mixing process in the nutrients or the contaminants behaviors requires the study of the physics of the flow. The mathematical models, used to describe the transport phenomena, promise to increase understanding about these behaviors.

In continuum mechanics, the basis of fluid dynamics models is the Navier-Stokes equations, that are nonlinear partial differential equations. The transport phenomena are described using the Navier-Stokes equations together with continuity equation and the scalar (such as temperature, salinity or water-soluble substance) transport equations. The nonlinear character of the Navier-Stokes equations makes it impossible to obtain the analytic solution of these in most cases. Improvements in computer power and storage, in recent decades, have enabled the

development of an area of fluid mechanics dedicated to the simulation of the flows employing computers (BATES; LANE; FERGUSON, 2005). This area is Computational Fluid Dynamics (CFD). CFD provides useful tools for the discretization of the mathematical models to obtain approximate solutions using scientific computation (WRIGHT, 2005).

There are several approaches to obtain the numerical solution of the fluid dynamics models. Classical techniques such as the Direct Numerical Simulation (DNS) can be used to simulate environmental flows, by solving the Navier-Stokes and scalar transport equations. DNS is very important for flow analysis because it gives details of the flow structures, however, this is possible only when the temporal and spatial resolution is fine enough to capture the whole range of scales of the flow (POPE, 2000). This requires large computational resources, which in practical engineering situations is not feasible.

Techniques based on statistical approach of Navier-Stokes equations provide a useful tool that is computationally viable for engineering purposes (SOTIROPOULOS, 2005). This approach relies on the Reynolds decomposition that consists in the separation of the instantaneous values variables into the time averaged component and the fluctuating component. Introducing the Reynolds decomposition into the fluid motion equations and applying time averages, the equations for the mean values variables are obtained. In this process new unknowns which are the average of products of fluctuations appear. These unknowns are described using the turbulence models.

The algebraic model is the simplest among the various models of turbulence. Its simplicity is that it is not necessary to solve any differential equation, and with the adjustments of a small amount of parameters realistic results can be obtained. The relevant dimensionless parameter is called Richardson number and involves the effects of inertia and the buoyancy effects. In stratified flow, the evaluation of this parameter becomes necessary to take into account the effect of interaction between turbulence and stratification.

The main goal of this work is to develop a numerical simulation tool for 3D environmental flow using an algebraic turbulence model based on the Richardson number. The approximation method used is the Finite Element Method (FEM). The Galerkin semi-discrete finite element method is used for spatial discretization. The nonlinear convective terms are discretized with the time derivative term. These terms are written in the form of substantial derivative, and are discretized by the semi-Lagrangian semi-implicit scheme. The MINI tetrahedral element



is adopted for velocity and pressure discretization and the linear tetrahedral element for scalar fields. These elements provide the consistent function spaces for the unknown functions that are necessary for simulation of incompressible fluid flows.

One advantage of using FEM, is that the FEM is based on unstructured meshes (FERZIGER; PERIĆ, 2002). A mesh composed of the simplices (triangles in 2D and tetrahedra in 3D) of Delaunay has the property to adapt to any complex geometry. This property is desirable to generate meshes that accurately represent complex natural topographies. Nowadays, there are several types of terrain data. And every tool for analysis of environmental flows must be able to read the various geographic data and generate a suitable mesh for the numerical simulation (FEM in this case). The main difficulty in mesh generation of tetrahedral finite element of the reservoir is non-uniform distribution of points. This non-uniform distribution is related to the huge ratio between the horizontal and vertical scales of the reservoir. In this type of points distribution, conventional tetrahedron mesh generation algorithm may become unstable. Hence, before expounding the numerical scheme, a tetrahedron mesh generation strategy to be used in numerical simulation is presented.

## **Objectives**

As mentioned above, the main objective of this work is to develop a numerical simulation tool for environmental flow using an algebraic turbulence model for stratified flow.

Specific objectives are:

- To briefly describe the types of terrain data employed in this work (chapter 1).
- To develop the tetrahedron mesh generation for FEM from the available terrain data (chapter 2).
- To describe the governing equations and the parameterizations of the variables inherent to the stratified turbulent flow (chapter 3).
- To describe the spatial Galerkin discretization and Semi-Lagrangian substantial derivative discretization (chapter 4).
- To develop a module of turbulence model for stratified flow and to incorporate into the existing FEM code of hydrodynamic transport simulation (chapter 5).
- To briefly describe the Object Oriented Programming (OOP) used to numerical sim-

ulation (chapter 5).

- To present the computational results of a simple model and to compare with the available literature (chapter 6).
- To present the simulation of the scale model of a confluence (chapter 6).
- To present the simulation of the flow in a branch of a reservoir (chapter 6).

# CHAPTER 1

---

## GEOGRAPHIC TOPOLOGY

---

Numerical modeling of hydrodynamic applications in real geophysical systems requires the correct representation of the terrain topological data. The main difficulty in the representation of the terrain data consist in the heterogeneity of the data sets that differ in precision, type, structure, among other factors. In this chapter, the different types of terrain data to be encountered in real geophysical systems and its features are presented.

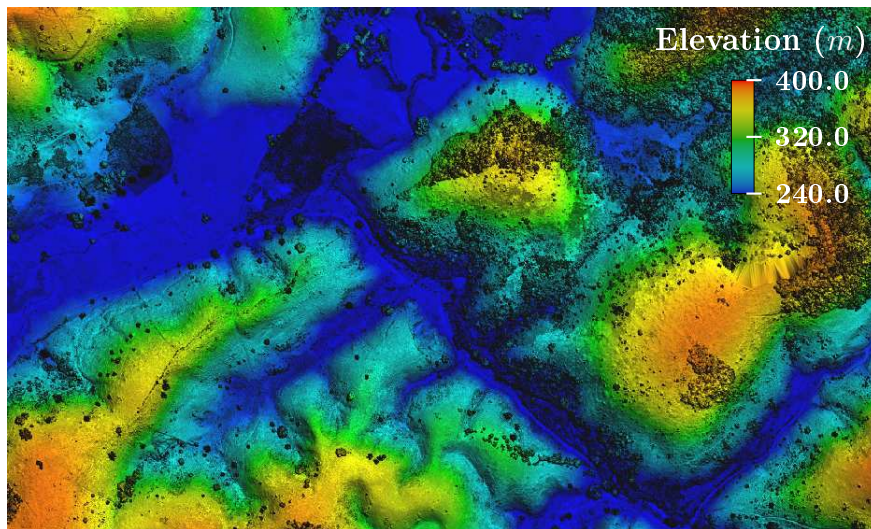
### **1.1 Raster and Vector data type**

There are two principal data models to represent the real world object. The first one is the raster data type, and another is the vector data type (HARMON; ANDERSON, 2003).

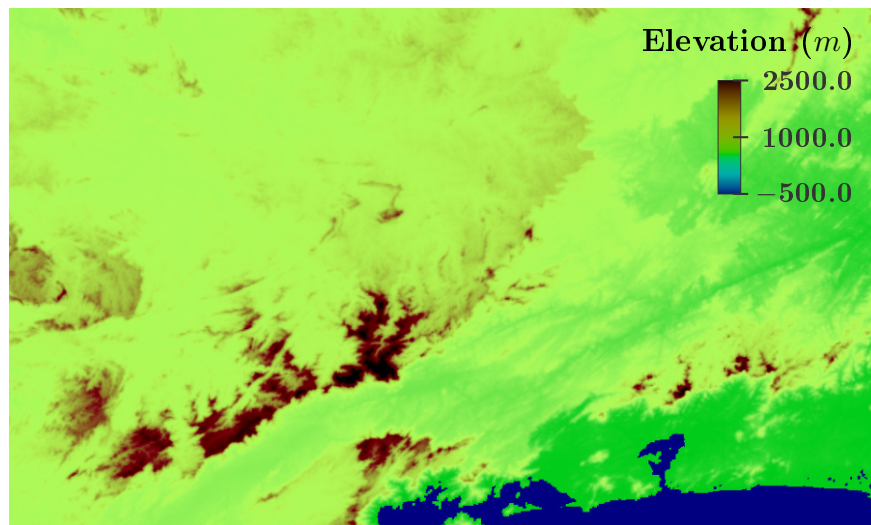
#### 1.1.1 Raster data type

The raster data type consists of rows and columns of cells, with each cell storing color value (RGB colors) or elevation (Figure 1.1). The raster which cells store color value is called raster images, and the cells are called pixel. Several file format such as TIFF, GeoTIFF, JPEG, are used to store raster images. The GeoTIFF file format embeds georeference information, such as coordinate systems, ellipsoids, datums, projections, within a TIFF file. Figure 1.1(a)

shows a GeoTIFF image of a region in *Rio de Janeiro* state (Brazil) close to  $43^{\circ}$  S –  $22^{\circ}$  W. The Digital Elevation Model (DEM) data file (also known as Digital Terrain Model (DTM)) stores the elevation information, and is used to represent the elevation as raster. Figure 1.1(b) shows a DTM image of the region of *Rio de Janeiro* state and south of *Minas Gerais* state in Brazil, between  $42^{\circ}$  and  $48^{\circ}$  S –  $21^{\circ}$  and  $24^{\circ}$  W.



(a)



(b)

Figure 1.1: Images of the raster data: (a) GeoTIFF image of a region in *Rio de Janeiro* state ( $43^{\circ}$  S –  $22^{\circ}$  W). (b) DTM image of the region of *Rio de Janeiro* state and south of *Minas Gerais* state.

### 1.1.2 Vector data type

In vector data type, the geographic information is expressed as vectors according to the geometrical shapes. The geospatial data are stored as points (simple location, such as, remote measure location), lines or polylines (such as rivers or contour lines), and polygons (such as lakes). Each shape (points/lines/polygons) has attributes that describes it, such as the name, elevation, visualization features and any meteorological or environmental measures. The shapefile (a collection of files with .shp, .shx, .dbf extension, developed and regulated by Environmental Systems Research Institute (ESRI)) is the common geographic vector data format (Figure 1.2). Figure 1.2(a) shows the image of a shpefile storing information of contour lines as polylines. The image corresponds to a region of *Nova Friburgo* in *Rio de Janeiro* state (Brazil) close to  $22.3^\circ$  S –  $42.6^\circ$  W. Figure 1.2(b) shows the image of some contour lines extracted from the shapefile showed in the Figure 1.2(a).

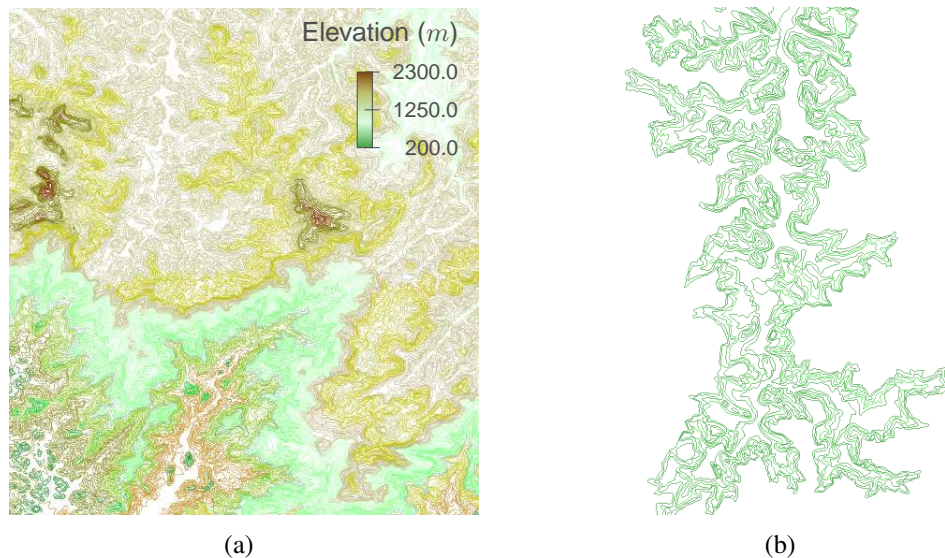


Figure 1.2: Images of the vector data: (a) Shapefile of a region of *Nova Friburgo* in *Rio de Janeiro* state. (b) Some contour lines of a river channel extracted from the shapefile showed in (a).

The shapefiles employed in this work have the information of the geometric data as polylines. These polylines form the contour lines. The cloud of points for the mesh generation is extracted from the contour lines. The Figure 1.3 shows the contour lines and the cloud of points lying on the contour lines.

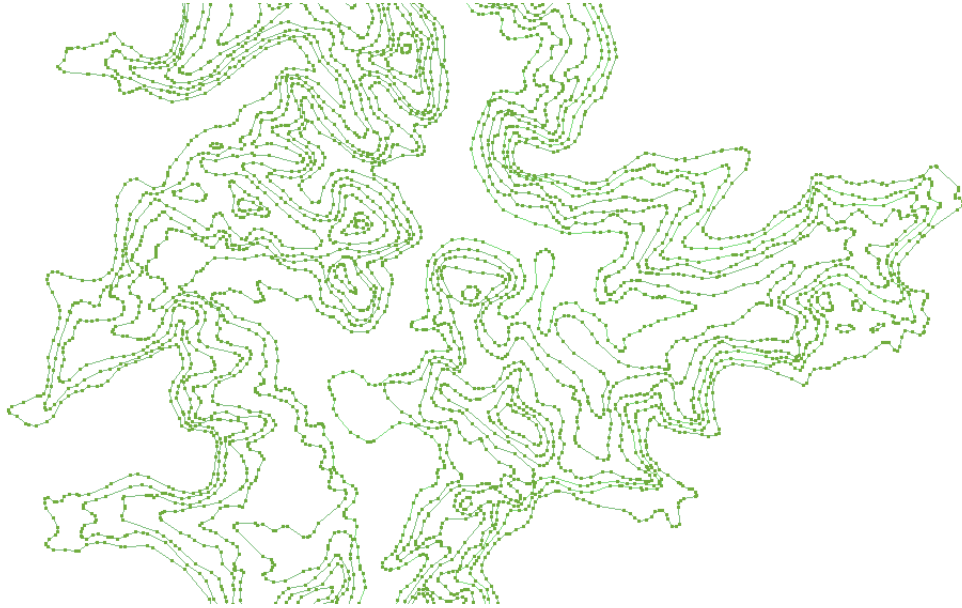


Figure 1.3: Image of the contour lines with the cloud of points.

## CHAPTER 2

---

### MESH GENERATION

---

This chapter describes the methodology used to discretize the spatial domain. From a cloud of points forming contour lines (shown in the previous chapter) of the continuous spatial domain, a surface mesh of Delaunay (BERG et al., 2000) triangles is generated using the incremental algorithm of nodes insertion. Then, a conforming mesh of tetrahedra is built. The non-uniform distribution of the points related to the huge ratio between the horizontal and vertical scales of the reservoir, commented in the introduction (chapter ), can be visualized in Figure 2.1. The properties of the Delaunay triangulation, the algorithm to generate it, and the process of building the tetrahedral mesh are presented.

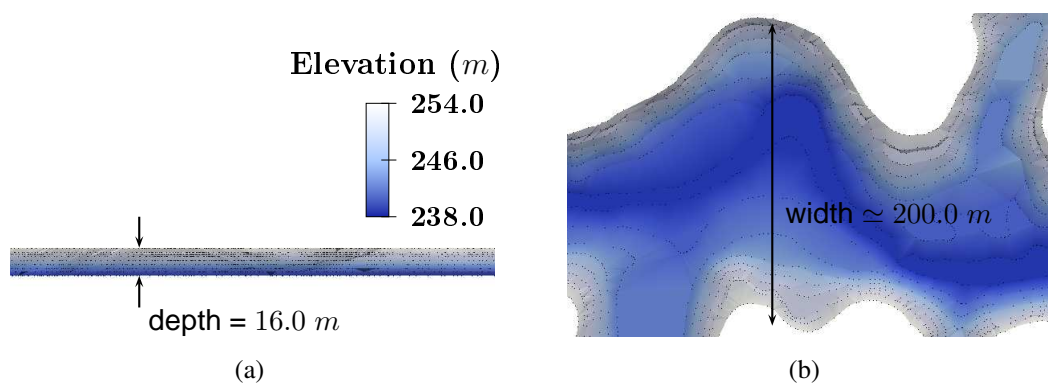


Figure 2.1: Non-uniform distribution of points in a river channel: (a) Side view. (b) Top view.

## 2.1 Delaunay triangulation

Given a set of points, the Delaunay triangulation maximizes the minimum angle of all possible triangulations (BERG et al., 2000; O'ROURKE, 1998). This property is desirable for the stiffness matrix conditioning in the FEM formulation (SHEWCHUK, 2002) and is equivalent to the property that the circumscribed circle (denoted as circumcircle) of any triangle in the Delaunay triangulation is empty (contains no other point in its interior).

### 2.1.1 Randomized incremental algorithm

A randomized incremental algorithm is used to compute a Delaunay triangulation. The point is inserted in random order, one at a time, and the triangulation is updated with each new addition. Consider a new point inserted into an initial triangulation (Figure 2.2(a)), triangles with circumcircle containing the new inserted point are located and deleted from the list of triangle. A new triangulation is obtained by joining the new point to the nodes at the edges of the cavity created by the previous removal of triangles (see Figure 2.2(b)).

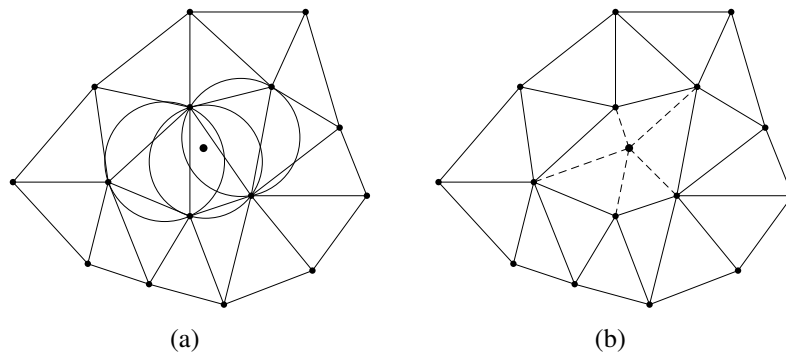


Figure 2.2: Randomized incremental algorithm of node insertion: (a) Given an initial triangulation, triangles with circumcircles containing a new point are removed. (b) A new triangulation is shown. (MAVRIPLIS, 1997)

Consider  $P = \{p_1, p_2, \dots, p_n\}$  be a set of points in the plane, the process begins with an initial triangulation formed by a single triangle containing all points of the set  $P$ . Let  $p_{-1}$ ,  $p_{-2}$  and  $p_{-3}$  the vertices of this triangle, the triangulation of the set  $P \cup \{p_{-1}, p_{-2}, p_{-3}\}$  is computed. After having inserted all points, the Delaunay triangulation of the set  $P$  is obtained by removing the points  $p_{-1}$ ,  $p_{-2}$  and  $p_{-3}$  with all incident edges.



### 2.1.2 Convex hull

To verify if a point lies in the circumcircle of a triangle, a geometry structure named convex hull is used. The convex hull of a set of points is the smallest convex set that contains the set of points. The lower convex hull of a set of points is defined as the portion of the convex hull that leaves all the points above it.

Let  $P' = \{p'_1, p'_2, \dots, p'_n\}$  be the projection of the points of  $P$  onto the paraboloid  $z = x^2 + y^2$ . The statement that the projection of the lower convex hull of  $P'$  onto the plane  $z = 0$  is the Delaunay triangulation of  $P$  will be shown.

The equation of the tangent plane to the paraboloid above some point  $(a, b, a^2 + b^2)$  is

$$z = 2ax + 2by - (a^2 + b^2) \quad (2.1)$$

and shifting this tangent plane upwards by some positive value  $r^2$  produces the plane whose equation is

$$z = 2ax + 2by - (a^2 + b^2) + r^2 \quad (2.2)$$

The equation obtained by intersecting this shifted plane and the paraboloid is

$$z = x^2 + y^2 = 2ax + 2by - (a^2 + b^2) + r^2 \quad (2.3)$$

$$(x - a)^2 + (y - b)^2 = r^2 \quad (2.4)$$

The equation (2.4) suggests that the projection of the intersection between the shifted plane and the paraboloid projected onto the plane  $z = 0$  is a circle with center  $(a, b, 0)$  and radius  $r$ . Notice that  $r^2$  is the distance that the plane has been shifted upwards from the point of tangency  $(a, b, a^2 + b^2)$ . It is easy to see that when the tangent plane to the paraboloid is shifting upwards the radius of the circle is growing. That is, all points of the paraboloid located below a shifted plane are projected on the plane  $z = 0$  inside the circle obtained by projecting on this plane (plane  $z = 0$ ) the intersection between the shifted plane and the paraboloid.

Now consider  $p_i, p_j,$  and  $p_k$  three points in the plane  $z = 0$ , and its projections  $p'_i, p'_j,$  and  $p'_k$  onto the paraboloid.  $p'_i, p'_j,$  and  $p'_k$  determine a plane  $\alpha$  that intersects the paraboloid. The projection of this intersection on the plane  $z = 0$  is a circle passing through  $p_i, p_j,$  and  $p_k$ .

Notice that the circle is the circumcircle of the triangle ( $\Delta p_i p_j p_k$ ) formed by the points  $p_i$ ,  $p_j$ , and  $p_k$ . By definition, the plane  $\alpha$  is one side of the lower convex hull if no other point is below this plane. And as shown in the previous paragraph, only the points below the plane  $\alpha$  have their projections on the plane  $z = 0$  inside the circle. That is, if there are no other point below the plane  $\alpha$ , then the circumcircle of the triangle  $\Delta p_i p_j p_k$  is empty, *i.e.*, the triangle  $\Delta p_i p_j p_k$  is a Delaunay triangle.

### 2.1.3 Non-uniform distribution of points

In a triangulation, it is possible that the edges of the triangles have a preferential direction. This is achieved by projecting onto an elliptic paraboloid, instead of projecting it onto a paraboloid of revolution. Given the direction and the ratio of non-uniformity, the equation of elliptic paraboloid is obtained from the quadratic form.

The equation of the paraboloid of revolution in the quadratic form is

$$z = x^2 + y^2 = \begin{pmatrix} x \\ y \end{pmatrix}^T \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} \quad (2.5)$$

The relation between the preferential direction and the original coordinate is given by two transformation matrices. Let  $\phi$  be the preferential direction measured counterclockwise, and  $\eta$  the ratio between the mean lengths of the edges in the preferential direction and the perpendicular direction to this. The  $(x, y)$  coordinate related to the preferential direction, let say  $(x_p, y_p)$ , using the transformation matrices is

$$\begin{pmatrix} x_p \\ y_p \end{pmatrix} = \begin{pmatrix} \cos \phi & -\sin \phi \\ \sin \phi & \cos \phi \end{pmatrix} \begin{pmatrix} \eta & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} \quad (2.6)$$

Rearranging this equation,

$$\begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} \frac{1}{\eta} \cos \phi & \frac{1}{\eta} \sin \phi \\ -\sin \phi & \cos \phi \end{pmatrix} \begin{pmatrix} x_p \\ y_p \end{pmatrix} \quad (2.7)$$

Substituting this equation into the equation (2.5), the following elliptic paraboloid equation is

obtained.

$$z = \left( \frac{1}{\eta^2} \cos^2 \phi + \sin^2 \phi \right) x_p^2 + \sin 2\phi \left( \frac{1}{\eta^2} - 1 \right) x_p y_p + \left( \frac{1}{\eta^2} \sin^2 \phi + \cos^2 \phi \right) y_p^2 \quad (2.8)$$

#### 2.1.4 Optimized search

A drawback for the Delaunay triangulation is its high computational cost as a new point is inserted. Notice that for each inserted point, to identify the triangles with circumcircle containing the point, it is necessary to search across all the triangles in the triangulation of the previous iteration. Hence, the searching algorithm is a bottleneck in this process.

To reduce the computational cost, an auxiliary structured quadrilateral mesh and a data structure with information of the neighbored triangles are used. With the auxiliary mesh, it is possible to recognize a triangle next to the new point inserted. The data structure reduces the search time to identify the triangles with circumcircle containing the point.

The disadvantage of using extra information for search is the increasing memory use. However, this information can be used to easily identify the nodes at the mesh boundary. This is important in order to impose the boundary conditions in the FEM routine.

## 2.2 Reservoir 3D Mesh Generation Routine

There are two features that are desirable and usually found in cartesian unstructured reservoir meshes:

**Layers.** The points are connected to other points at the same  $z$  coordinate. This feature is desirable, specially in strongly stratified flows since the velocity is preferably horizontal, and a mesh constructed with horizontal layers will introduce less artificial diffusion.

**Sticks.** The points are connected vertically to other points at the same  $x, y$  coordinate, but at a different  $z$  coordinate. This feature is desirable because it allows a simplification of the pressure calculation, which varies almost hydrostatically in the vertical direction. In the limit of very low vertical accelerations, the pressure distribution can be assumed hydrostatic and pressure can be computed from the surface pressure, integrating vertically the  $z$ -momentum equation.

Even if the pressure distribution is not assumed to be hydrostatic, the vertical alignment in the sticks produces a better coupling between the vertical velocity and the pressure fields.

The routine of mesh generation employs as input data the coordinate points of the contour lines corresponding to the bottom of the reservoir. However, these points are not sufficient to generate a tetrahedral mesh. Therefore additional points need to be defined inside the reservoir. Next, the mechanism of inclusion of these additional points producing layers and sticks, is discussed.

### 2.2.1 Points adding process

Throughout each point that does not belong to the upper (external) contour lines, a vertical line (stick) is drawn. This vertical lines intersect the horizontal planes (layers). The intersections lying inside the reservoir are the points to be added to the set of points  $P$ . Notice that now the points are classified in layers. The Figure 2.3(a) shows an example of the contour lines points, while the Figure 2.3(b) shows the points after the point adding process.

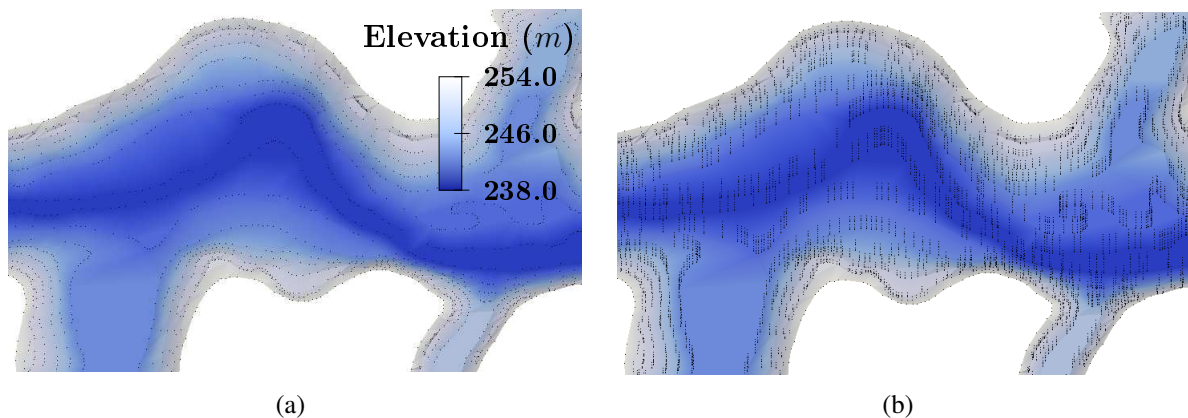


Figure 2.3: Points adding process: (a) contour lines points. (b) points filling the reservoir.

### 2.2.2 Prism partition process

After the point adding process, the code generates a two-dimensional mesh using the Delaunay triangulation with the information of the sticks and layers. Once the Delaunay triangulation was performed, the triangular prisms are constructed and partitioned in order to produce the tetrahedral elements.

The process of dividing the prisms is not trivial since it is possible to obtain a non-conforming tetrahedron (see Figure 2.4(a)). Therefore, restrictions during the prism partitioning process are imposed.

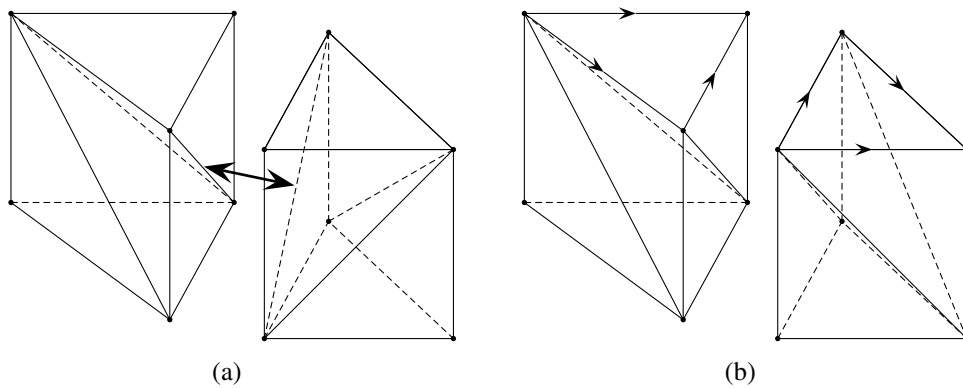


Figure 2.4: Partitioning prisms: (a) non-conforming tetrahedron. (b) conforming tetrahedron.

To solve this problem, each edge is orientated (using arrows) considering the crescent  $x$ -coordinate. Using the origin vertex of the arrow as a starting point, diagonals are drawn to each lateral face (see Figure 2.4(b)). Notice that, since the orientation of the edges is arbitrary, the partition of the prism is not unique.

A 3D mesh is shown in Figure 2.5. It can be observed that the final mesh has the desired feature: the points lie on the intersection of planes (layers) with vertical lines (sticks).

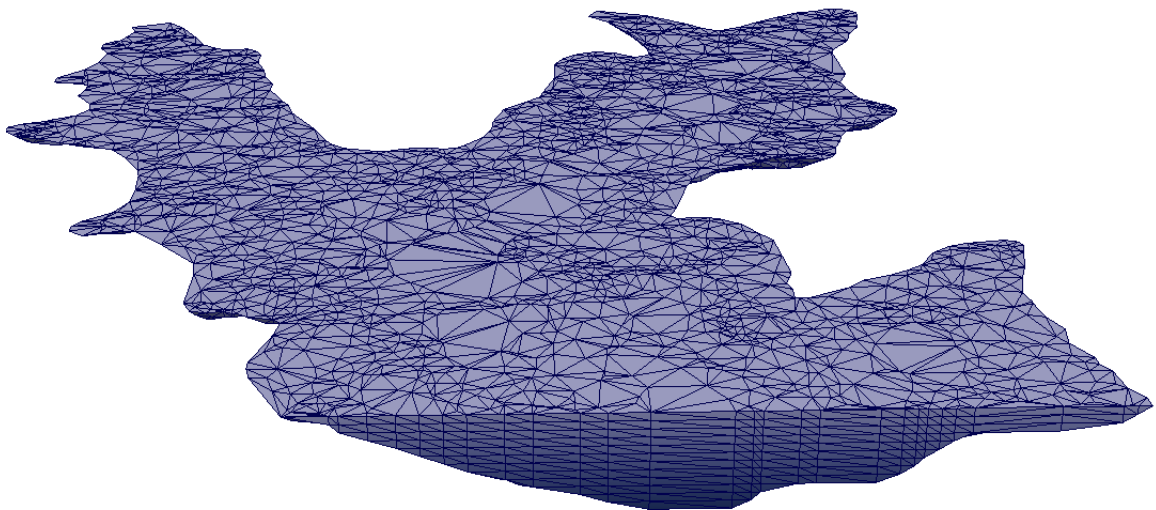


Figure 2.5: Three-dimensional Mesh.

## CHAPTER 3

---

# GOVERNING EQUATIONS

---

This chapter describes the derivation of the Reynolds Averaged Navier-Stokes (RANS) and mean scalar transport equations, and then analyzes the eddy viscosity/diffusivity model, specifically the algebraic model applied to numerical simulation. At last, the stratified turbulent flow model is shown. To represent the equations, index notation has been used.

### 3.1 Fundamental equations of fluid flow

The fundamental equations of Fluid Mechanics are derived from the conservation laws of mass and *momentum* (BATCHELOR, 2000; KUNDU; COHEN, 2002). The continuity, Navier-Stokes and scalar transport equations based on Boussinesq approximation and Reynolds averaging are shown.

#### 3.1.1 Continuity equation

The conservation of mass leads to the continuity equation, and reads:

$$\frac{1}{\rho} \frac{D\rho}{Dt} + \frac{\partial u_i}{\partial x_i} = 0 \quad (3.1)$$

where,  $u_i (i = 1, 2, 3)$  are the instantaneous velocity components,  $x_i (i = 1, 2, 3)$  are the coordinate axes with direction 3 vertically upward,  $\rho$  is the density of the fluid, and the substantive derivative operator  $D/Dt$  is defined as

$$\frac{D}{Dt} = \frac{\partial}{\partial t} + u_i \frac{\partial}{\partial x_i} \quad (3.2)$$

Based on the Boussinesq approximation, the continuity equation (3.1) reduces to the incompressible form:

$$\frac{\partial u_i}{\partial x_i} = 0 \quad (3.3)$$

The Boussinesq approximation states that density differences at low Mach numbers (defined as  $Mc = U/c$ , where  $U$  is a characteristic velocity of the flow and  $c$  is the sound velocity in the fluid) are sufficiently small and can be neglected, except in the gravity term where  $\rho$  is multiplied by  $g_i (i = 3)$ , the acceleration due to gravity (KUNDU; COHEN, 2002; OERTEL, 2004).

### 3.1.2 Navier-Stokes equation

The law of *momentum* conservation relates the acceleration of fluid particles with the forces exerted on them. Two types of forces are considered: body forces, and surface forces. The body forces are related to the gravity forces and the surface forces, which have a molecular origin, are represented by the stress tensor  $\sigma_{ij} (i, j = 1, 2, 3)$ . Then, the conservation of the linear *momentum* is given by the equation:

$$\rho \frac{Du_i}{Dt} = \frac{\partial \sigma_{ij}}{\partial x_j} + \rho g_i \quad (3.4)$$

The conservation of angular *momentum* shows that the stress tensor is symmetric. For a fluid in motion, the stress tensor is split into a part  $-p\delta_{ij}$  (where,  $p$  is the pressure of the fluid and  $\delta_{ij} (i, j = 1, 2, 3)$  is the Kronecker delta) that would exist if the fluid were at rest and additional components  $\tau_{ij} (i, j = 1, 2, 3)$  (called the deviatoric stress tensor) due to viscosity:

$$\sigma_{ij} = -p\delta_{ij} + \tau_{ij} \quad (3.5)$$

Newtonian fluids are those in which there is a linear relation between deviatoric stress and rate of strain tensor, *i.e.*,

$$\tau_{ij} = \mu \left( \frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right) \quad (3.6)$$

where the constant of proportionality  $\mu$  is the molecular dynamic viscosity of the fluid (the kinematic viscosity is defined as  $\nu = \mu/\rho$ ). Then, for Newtonian fluids, the equation (3.5) takes the form

$$\sigma_{ij} = -p\delta_{ij} + \mu \left( \frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right) \quad (3.7)$$

Substitution of equation (3.7) into the *momentum* equation (3.4) gives

$$\rho \frac{Du_i}{Dt} = -\frac{\partial p}{\partial x_i} + \frac{\partial}{\partial x_j} \left[ \mu \left( \frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right) \right] + \rho g_i \quad (3.8)$$

This equation is called the Navier-Stokes (NS) equation.

Consider the case where the fluid is at rest and the density is uniform throughout the domain. Let  $\rho_r$  be the density of reference and  $p_r$  the reference pressure. Then,

$$\frac{\partial p_r}{\partial x_i} = \rho_r g_i \quad (3.9)$$

Subtracting this equation from equation (3.8) and dividing by  $\rho_r$ , it is obtained

$$\frac{\rho}{\rho_r} \frac{Du_i}{Dt} = -\frac{1}{\rho_r} \frac{\partial(p - p_r)}{\partial x_i} + \frac{1}{\rho_r} \frac{\partial}{\partial x_j} \left[ \mu \left( \frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right) \right] + \frac{\rho - \rho_r}{\rho_r} g_i \quad (3.10)$$

Relying on the Boussinesq approximation (KUNDU; COHEN, 2002; OERTEL, 2004), the density difference is considered negligible in the term of left hand side, *i.e.*,  $\rho/\rho_r = 1$ . However, the density difference in the term that contains the acceleration due to gravity remains.

Finally, the Navier-Stokes equation for incompressible and Newtonian fluid with Boussinesq approximation is

$$\frac{Du_i}{Dt} = -\frac{1}{\rho_r} \frac{\partial p}{\partial x_i} + \frac{1}{\rho_r} \frac{\partial}{\partial x_j} \left[ \mu \left( \frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right) \right] + \frac{\rho - \rho_r}{\rho_r} g_i \quad (3.11)$$

where  $p$  is the pressure variation from reference pressure.



### 3.1.3 Scalar transport equation

Let  $\theta$  be a scalar quantity of magnitude of a property of the fluid (*e.g.* temperature) or concentration of a substance (like salt) that induces stratification in the flow. The transport equation of scalar quantity  $\theta$ , is governed by (KUNDU; COHEN, 2002; BATCHELOR, 2000; LANDAU; LIFSHITZ, 1987)

$$\frac{D\theta}{Dt} = \frac{\partial}{\partial x_j} \left( \alpha \frac{\partial \theta}{\partial x_j} \right) + S_\theta \quad (3.12)$$

where  $S_\theta$  is a source or sink of  $\theta$  and  $\alpha$  is the molecular (heat or mass) diffusivity coefficient.

Relying on the Boussinesq approximation, in equation (3.12), the scalar quantity  $\theta$  can be considered as a variation with respect to a reference value  $\theta_r$ .

The local density of the equation (3.11) is related to  $\theta$  via an equation of state of the following form:

$$\frac{\rho - \rho_r}{\rho_r} = -\beta\theta \quad (3.13)$$

where,  $\beta$  is the thermal expansion coefficient in case the local density is related to the temperature.

## 3.2 Derivation of mean equations

The statistical approach is used to simulate turbulent flow. Turbulent flows are characterized by the existence of several length scales. This approach acts as a filter that removes all small length scales in the flow. The statistical approach based on Reynolds decomposition is shown. Then, the fundamental equations are derived for the mean equation of turbulent flow.

### 3.2.1 Statistical approach

The statistical approach is based on Reynolds decomposition. Instantaneous value of each dependent variable is decomposed as a sum of a mean and a fluctuation around this mean value. For example, for the component  $i$  of velocity,

$$u_i(\mathbf{x}, t) = U_i(\mathbf{x}, t) + u'_i(\mathbf{x}, t) \quad (3.14)$$

where  $U_i(\mathbf{x}, t)$  is the mean part and  $u'_i(\mathbf{x}, t)$  is the random fluctuation part. The mean value is given by

$$U_i(\mathbf{x}, t) = \frac{1}{T} \int_t^{t+T} u_i(\mathbf{x}, t) dt \quad (3.15)$$

where the averaging interval  $T$  is taken to be much longer than the maximum period of the turbulent fluctuations. Notice that

$$\overline{u'_i(\mathbf{x}, t)} = 0 \quad (3.16)$$

where the overbar denotes time averaging.

### 3.2.2 Mean continuity equation

Substituting the Reynolds decomposition, equation (3.14), into the continuity equation (3.3)

$$\begin{aligned} \frac{\partial(U_i + u'_i)}{\partial x_i} &= 0 \\ \frac{\partial U_i}{\partial x_i} + \frac{\partial u'_i}{\partial x_i} &= 0 \end{aligned} \quad (3.17)$$

Averaging this equation and taking into account the equation (3.16), the continuity equation for mean motion takes the form:

$$\frac{\partial U_i}{\partial x_i} = 0 \quad (3.18)$$

Notice that the mean continuity equation (equation 3.18) is identical to the instantaneous equation (equation 3.3), with the instantaneous velocity replaced by the mean velocity.

Subtracting the equation (3.18) from the equation (3.17), the continuity equation for the fluctuations is obtained

$$\frac{\partial u'_i}{\partial x_i} = 0 \quad (3.19)$$

### 3.2.3 Reynolds averaged Navier-Stokes equations

To derive the mean equation of NS equations, the equation (3.11) is rewritten, considering the equality (3.13), as follows

$$\frac{\partial u_i}{\partial t} + \frac{\partial(u_i u_j)}{\partial x_j} = -\frac{1}{\rho_r} \frac{\partial p}{\partial x_i} + \frac{1}{\rho_r} \frac{\partial}{\partial x_j} \left[ \mu \left( \frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right) \right] - \beta \theta g_i \quad (3.20)$$

Applying the Reynolds decomposition, the above equation reads

$$\begin{aligned} & \frac{\partial(U_i + u'_i)}{\partial t} + \frac{\partial((U_i + u'_i)(U_j + u'_j))}{\partial x_j} \\ &= -\frac{1}{\rho_r} \frac{\partial(P + p')}{\partial x_i} + \frac{1}{\rho_r} \frac{\partial}{\partial x_j} \left[ \mu \left( \frac{\partial(U_i + u'_i)}{\partial x_j} + \frac{\partial(U_j + u'_j)}{\partial x_i} \right) \right] - \beta(\Theta + \theta') g_i \end{aligned} \quad (3.21)$$

where  $P$  is defined as

$$P(\mathbf{x}, t) = \frac{1}{T} \int_t^{t+T} p(\mathbf{x}, t) dt \quad (3.22)$$

and,  $\Theta$  is given by

$$\Theta(\mathbf{x}, t) = \frac{1}{T} \int_t^{t+T} \theta(\mathbf{x}, t) dt \quad (3.23)$$

Averaging the equation (3.21), the Reynolds averaged Navier-Stokes (RANS) equations are obtained,

$$\frac{\partial U_i}{\partial t} + \frac{\partial(U_i U_j + \overline{u'_i u'_j})}{\partial x_j} = -\frac{1}{\rho_r} \frac{\partial P}{\partial x_i} + \frac{1}{\rho_r} \frac{\partial}{\partial x_j} \left[ \mu \left( \frac{\partial U_i}{\partial x_j} + \frac{\partial U_j}{\partial x_i} \right) \right] - \beta \Theta g_i \quad (3.24)$$

Defining the mean substantial derivative as

$$\frac{\overline{D}}{Dt} = \frac{\partial}{\partial t} + U_i \frac{\partial}{\partial x_i} \quad (3.25)$$

the equation (3.24) can be written as follows

$$\frac{\overline{D}U_i}{Dt} = -\frac{1}{\rho_r} \frac{\partial P}{\partial x_i} + \frac{1}{\rho_r} \frac{\partial}{\partial x_j} \left[ \mu \left( \frac{\partial U_i}{\partial x_j} + \frac{\partial U_j}{\partial x_i} \right) - \rho_r \overline{u'_i u'_j} \right] - \beta \Theta g_i \quad (3.26)$$

The correlation  $\overline{u'_i u'_j}$  appears when averaging the product of two quantities ( $u_i$  and  $u_j$ ). The tensor  $-\rho_r \overline{u'_i u'_j}$  due to the fluctuation of the velocity is known as the Reynolds stress tensor.

This tensor is symmetric, and thus has six independent components. Also, the tensor  $\overline{u'_i u'_j}$  is referred to Reynolds tensor.

### 3.2.4 Mean scalar transport equation

Proceeding in a similar way to the previous section, the equation (3.12) is rewritten as follows

$$\frac{\partial \theta}{\partial t} + \frac{\partial (u_j \theta)}{\partial x_j} = \frac{\partial}{\partial x_j} \left( \alpha \frac{\partial \theta}{\partial x_j} \right) + S_\theta \quad (3.27)$$

Applying the Reynolds decomposition, the equation (3.27) reads

$$\frac{\partial (\Theta + \theta')}{\partial t} + \frac{\partial ((U_j + u'_j)(\Theta + \theta'))}{\partial x_j} = \frac{\partial}{\partial x_j} \left( \alpha \frac{\partial (\Theta + \theta')}{\partial x_j} \right) + S_\theta \quad (3.28)$$

Averaging this equation, the mean scalar equation is obtained, which using the equation (3.25), has the following form

$$\frac{\overline{D}\Theta}{Dt} = \frac{\partial}{\partial x_j} \left( \alpha \frac{\partial \Theta}{\partial x_j} - \overline{\theta' u'_j} \right) + S_\theta \quad (3.29)$$

The correlation  $\overline{\theta' u'_j}$  is a vector called turbulent scalar flux.

### 3.2.5 The closure problem

For a three dimensional turbulent transport, there are five independent equations given by the three components of RANS equations (equation 3.26), mean continuity equation (equation 3.18) and mean scalar transport equation (equation 3.29). However these five equations contain more than five unknowns. In addition to  $U_i (i = 1, 2, 3)$ ,  $P$  and  $\Theta$  (five quantities), the Reynolds stress tensor introduces six unknowns and the scalar flux three additional unknowns.

Consequently the Reynolds equations system is unclosed. That is, the system cannot be solved except when the Reynolds stress and turbulent scalar flux are determined. For this purpose, the Boussinesq eddy viscosity/diffusivity approach is used.

### 3.3 Eddy-viscosity/diffusivity model

The turbulent kinetic energy is defined as (POPE, 2000):

$$k = \frac{1}{2} \rho_r \overline{u_i'^2} \quad (3.30)$$

That is half of the trace of the Reynolds tensor. As the Reynolds tensor is symmetrical, there are axes (called principal axes), for which the tensor is diagonal. The diagonal components are the eigenvalues  $\lambda_i$  ( $i = 1, 2, 3$ ) of the tensor.

The isotropic part of the tensor is defined as a tensor with diagonal components equal to

$$\frac{\lambda_i}{3} = \frac{2}{3} \rho_r k \quad (3.31)$$

and their off-diagonal components are zero. Thus, the isotropic part of the Reynolds tensor reads

$$\frac{2}{3} \rho_r k \delta_{ij} \quad (3.32)$$

The deviatoric part of the Reynolds tensor, defined as the difference between the Reynolds tensor and its isotropic part, which reads

$$-\rho_r \overline{u_i' u_j'} + \frac{2}{3} \rho_r k \delta_{ij} \quad (3.33)$$

In analogy to the relationship between the deviatoric stress and the strain rate tensor of Newtonian fluids (equation 3.6), J. Boussinesq has suggested that the deviatoric Reynolds stress tensor (equation 3.33) is proportional to the mean strain rate tensor, with the proportionality factor  $\mu_t$  called eddy viscosity (POPE, 2000; WILCOX, 1993). The hypothesis reads

$$-\rho_r \overline{u_i' u_j'} + \frac{2}{3} \rho_r k \delta_{ij} = \rho_r \mu_t \left( \frac{\partial U_i}{\partial x_j} + \frac{\partial U_j}{\partial x_i} \right) \quad (3.34)$$

Incorporating this assumption in the average Reynolds equation (3.26),

$$\frac{\overline{D}U_i}{Dt} = -\frac{1}{\rho_r} \frac{\partial}{\partial x_i} \left( P + \frac{2}{3} \rho_r k \right) + \frac{1}{\rho_r} \frac{\partial}{\partial x_j} \left[ \mu_{eff} \left( \frac{\partial U_i}{\partial x_j} + \frac{\partial U_j}{\partial x_i} \right) \right] - \beta \Theta g_i \quad (3.35)$$

where

$$\mu_{eff}(\mathbf{x}, t) = \mu + \mu_t(\mathbf{x}, t) \quad (3.36)$$

is the effective viscosity. While the molecular viscosity  $\mu$  (property of the fluid) is considered constant throughout the domain, the eddy viscosity  $\mu_t$  is not a physical property, and depends on the flow under consideration.

It is possible to obtain a kinematic effective viscosity  $\nu_{eff}$ , dividing the effective viscosity  $\mu_{eff}$  by the density of the fluid  $\rho_r$ . The expression  $P + \frac{2}{3}\rho_r k$  of the equation (3.35) is a modified mean pressure and can be considered simply as  $P$ . Then, the equation (3.35) is rewritten in terms of the kinematic effective viscosity  $\nu_{eff}$  and the modified mean pressure  $P$  as

$$\frac{\overline{D}U_i}{Dt} = -\frac{1}{\rho_r} \frac{\partial P}{\partial x_i} + \frac{\partial}{\partial x_j} \left[ \nu_{eff} \left( \frac{\partial U_i}{\partial x_j} + \frac{\partial U_j}{\partial x_i} \right) \right] - \beta \Theta g_i \quad (3.37)$$

Similarly, in analogy to Fourier's law of the heat transfer or Fick's law of molecular diffusion, the turbulent scalar flux is proportional to the mean scalar gradient, with the proportionality factor  $\alpha_t$  called eddy or turbulent diffusivity. The hypothesis reads

$$-\overline{\theta' u'} = \alpha_t \frac{\partial \Theta}{\partial x_j} \quad (3.38)$$

Substituting into the equation (3.29)

$$\frac{\overline{D}\Theta}{Dt} = \frac{\partial}{\partial x_j} \left( \alpha_{eff} \frac{\partial \Theta}{\partial x_j} \right) + S_\theta \quad (3.39)$$

where

$$\alpha_{eff}(\mathbf{x}, t) = \alpha + \alpha_t(\mathbf{x}, t) \quad (3.40)$$

is the effective diffusivity.

### 3.4 Algebraic models

Algebraic models are zero (differential) equation models to estimate the eddy-viscosity diffusivity. The algebraic models rely on Prandtl's mixing length hypothesis (OERTEL, 2004; SCHLICHTING; GERSTEN, 2000). Consider the velocity profile in the boundary layer of a two-

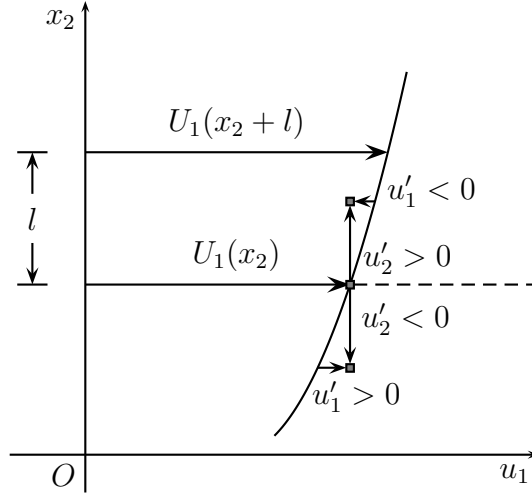


Figure 3.1: Prandtl's mixing length: average velocity profile in the boundary layer of a two-dimensional flow.

dimensional flow, shown in the figure 3.1. To simplify, it is assumed that the average velocity  $U_1$  is dependent only on  $x_2$ . Assuming that a fluid element has moved from a position  $x_2$  with average velocity  $U_1$  by distance  $l$ , the order of magnitude of the fluctuating velocity  $u'_1$ , in the  $x_1$  direction, is given by the difference in velocity that the particle has suffered

$$\Delta U_1 = U_1(x_2 + l) - U_1(x_2) \quad (3.41)$$

Expanding  $U_1(x_2 + l)$  in a Taylor series and ignoring higher order terms, yields

$$\Delta U_1 = l \frac{\partial U_1}{\partial x_2} \quad (3.42)$$

Based on the continuity equation for the fluctuations (equation 3.19), the fluctuation of velocity  $u'_2$ , in the  $x_2$  direction, is the same order of magnitude as that of  $u'_1$ . Then, the Reynolds stress tensor is of order of magnitude

$$\rho_r \left( l \frac{\partial U_1}{\partial x_2} \right)^2 \quad (3.43)$$

The sign of the Reynolds stress tensor  $-\rho_r \overline{u'_1 u'_2}$  is positive, since  $u'_1$  and  $u'_2$  have opposite signs and their product  $u'_1 u'_2$  is negative. To ensure that the Reynolds stress tensor has the same sign

that the velocity gradient, it is expressed as follows

$$-\rho_r \overline{u'_1 u'_2} = \rho_r l^2 \left| \frac{\partial U_1}{\partial x_2} \right| \frac{\partial U_1}{\partial x_2} \quad (3.44)$$

Comparing this equation with the equation of the eddy viscosity, the following relation for kinematic eddy viscosity is obtained

$$\nu_t = l^2 \left| \frac{\partial U_1}{\partial x_2} \right| \quad (3.45)$$

As a generalization for 3D flow, the equation (3.45) is expressed as

$$\nu_t = 2l^2 \sqrt{S_{ij} S_{ij}} \quad (3.46)$$

where  $S_{ij}$  is the mean strain rate tensor defined as

$$S_{ij} = \frac{1}{2} \left( \frac{\partial U_i}{\partial x_j} + \frac{\partial U_j}{\partial x_i} \right) \quad (3.47)$$

The length  $l$ , called the mixing length, is a characteristic turbulence length and represents the average diameter of the eddies. Notice that a solid surface has an effect on the size of the vortices. This means that a solid surface has an influence on the mixing length. Therefore, the influences of the solid surfaces in the flow must be taken into account. The part of the fluid mechanics that studies the effect of solid surface in the flow is called the boundary layer theory. A brief description of the boundary layer theory is shown in the following section.

### 3.5 Boundary layer

The fluid flow is influenced by the presence of solid surfaces. Fluid particles immediately next to the surface remain at rest, a situation called no-slip condition. And there is a transition zone for the velocity (nonzero velocity gradient) near the surface. This zone is called boundary layer. In the boundary layer, the flow is dominated mainly by viscous effects. Outside the boundary layer, there is a region called inviscid outer flow where the viscous effects can be neglected.

Depending on the type of flow there are two types of boundaries layer: the laminar



boundary layer and turbulent boundary layer. A parameter characterizes the condition of flow. This parameter is the Reynolds number ( $Re$ ), which is a dimensionless number: ratio of inertial forces and viscous forces, defined as

$$Re = \frac{UL}{\nu} \quad (3.48)$$

where  $U$  and  $L$  are characteristic velocity and some characteristic length of the flow respectively. For low Reynolds numbers the flow is laminar. The transition from laminar to turbulent occurs above a certain value of the Reynolds number. This value is called the critical Reynolds number ( $Re_{crit}$ ). The laminar-turbulent transition is affected by various factors including the nature of the surface (roughness), the pressure distribution of the outer flow and turbulence intensity.

The turbulent boundary layer is split into three regions: the inner region (viscous wall region); the log-law region and the outer region (core layer). The inner region, where the size of the vortices depends on the proximity of the surface, is called viscous wall region. Within the viscous wall region there are two layers. The viscous sublayer is the part of the wall viscous region close to the wall. In this layer the Reynolds stress tensor is negligible compared to the viscous stress tensor (POPE, 2000). The layer where the effects of turbulent and viscous effects are of the same order of magnitude is the buffer layer. The log-law region is the transition region between the viscous wall region and the outer region. The name log-law is because in this region the velocity profile is logarithmic. In the log-law region the production and the dissipation of the turbulent energy are almost in balance. In the core layer (outer region), the flow is dominated by inertia effects. In this region the size of vortices is constant.

The figure 3.2 shows the mixing length model for turbulent boundary layer.  $\delta$  is the thickness of the boundary layer, defined as the distance to the surface where the velocity is 99% of the velocity of the outer flow.  $\kappa = 0.41$  and  $\lambda = 0.09$  are constants.  $\kappa$  is called von Kármán constant. Viscous wall region and log-law region are restricted to values of  $x_3$  less than  $(\lambda/\kappa)\delta$  (about fifth of the boundary layer). In this zone the mixing length is proportional to the distance to the surface and in the outer region the mixing length is constant.

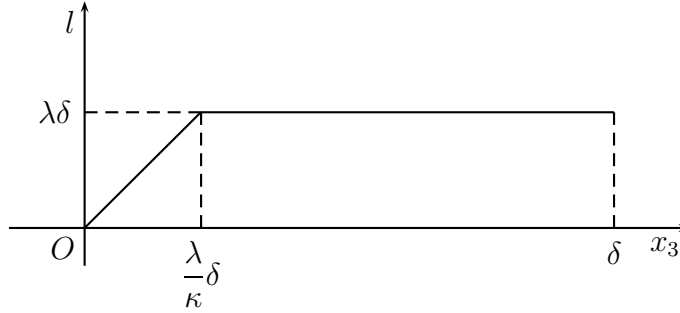


Figure 3.2: Mixing length model for turbulent boundary layer.

This model reads

$$l = \begin{cases} \kappa x_3 & 0 < x_3 \leq \frac{\lambda}{\kappa} \delta \\ \lambda \delta & \frac{\lambda}{\kappa} \delta \leq x_3 \leq \delta \end{cases} \quad (3.49)$$

### 3.6 Turbulence modeling for stratified flows

The interaction between turbulence and stratification is parameterized in terms of the local gradient Richardson number. The local gradient Richardson number is defined as

$$Ri = \frac{N_L^2}{\left(\frac{\partial U}{\partial z}\right)^2} \quad (3.50)$$

where  $N_L$  is the local Brunt-Väisälä buoyancy frequency and  $z$  is the direction of stratification.

The Brunt-Väisälä frequency is defined as

$$N_L = \left(-\frac{g}{\rho_r} \frac{\partial \rho}{\partial z}\right)^{\frac{1}{2}} \quad (3.51)$$

The Brunt-Väisälä frequency gives the stabilizing influence of the stratification. If  $N_L^2 > 0$ , then the Brunt-Väisälä frequency is the frequency of oscillation due to the buoyancy effects within a stable environment. The stratification is unstable if  $N_L^2 < 0$ .

The denominator of the equation (3.50) is the vertical shear in the mean horizontal velocity and gives the destabilizing influence of the shear.

Since the sign of the denominator of the Richardson number is always positive, the sign of the Richardson number depends only on the Brunt-Väisälä frequency. Negative Richardson number corresponds to the case where both buoyancy and shear destabilize the flow generating

turbulence. For a stable environment, the  $Ri$  is positive.

From a critical value of Richardson number, denoted as  $Ri_c$ , turbulent motion is suppressed by the stable density gradient. When  $Ri < Ri_c$ , the buoyancy is unable to suppress the turbulence, and the turbulent viscosity depends on the  $Ri$ . Sotiropoulos (2005) suggests a modified mixing-length model for stratification. This model can be formulated as follows (SOTIROPOULOS, 2005):

Stable stratification ( $Ri > 0$ )

$$\nu_t = \begin{cases} 2l^2 \sqrt{S_{ij}S_{ij}} (1 - Ri/Ri_c)^2 & 0 \leq Ri \leq Ri_c \\ 0 & Ri \geq Ri_c \end{cases} \quad (3.52)$$

Unstable stratification

$$\nu_t = 2l^2 \sqrt{S_{ij}S_{ij}} (1 - Ri)^{1/2} \quad (3.53)$$

### 3.7 Boundary and initial condition

The formulation of a problem modeled by differential equations is given by a set of differential equations whose solutions satisfy certain restrictions so-called boundary and initial conditions. There are two types of boundary conditions used. The Dirichlet boundary conditions describe the variables values along the boundary of the domain and the Neumann boundary conditions describe the variables gradients along the boundary. Initial conditions are specified by the variables values throughout the domain at time  $t = t_0$ .

Four type of boundary types are mentioned. The  $t - n$  coordinate system is used. The  $t$  and  $n$  have the direction of the tangent and normal, respectively, to the surface in each point.

**Solid walls boundary conditions.** On solid walls (impermeable surface) the no-slip condition is applied, *i.e.*

$$U_t = U_n = 0 \quad (3.54)$$

**Free surface symmetry plane conditions.** Dirichlet boundary condition for velocity normal to the surface with zero value is employed. For normal gradients of the tangential velocity

and scalar concentration are zero (Neumann boundary condition).

$$U_n = \frac{\partial U_t}{\partial x_n} = \frac{\partial \Theta}{\partial x_n} = 0 \quad (3.55)$$

**Inflow conditions.** At inflow surface, the values of the velocity and the scalar concentration are prescribed for each simulation of interest. For example, the following definition is possible

$$U_n = U_{inflow}; \quad U_t = 0; \quad \Theta = \Theta_{inflow} \quad (3.56)$$

where  $U_{inflow}$  and  $\Theta_{inflow}$  are values defined for each simulation.

**Outflow conditions.** At outflow surface, it is possible to assume that the flow is fully developed. Therefore, the Neumann condition for the velocity and the scalar concentration is employed.

$$\frac{\partial U_n}{\partial x_n} = \frac{\partial U_t}{\partial x_n} = \frac{\partial \Theta}{\partial x_n} = 0 \quad (3.57)$$

### 3.8 Nondimensional form

The equations (3.18), (3.37) and (3.39) are written in nondimensional form. For this, the following scales for the variables are used:

$$\begin{aligned} U_i^* &= \frac{U_i}{U_c} & x_i^* &= \frac{x_i}{L_c} & \rho^* &= \frac{\rho}{\rho_r} & P^* &= \frac{P}{\rho_r U_c^2} \\ t^* &= \frac{t U_c}{L_c} & \Theta^* &= \frac{\Theta}{\Theta_r} & g_i^* &= \frac{g_i}{g_r} \end{aligned}$$

where,  $U_c$  is the characteristic velocity of the flow,  $L_c$  is the characteristic length and  $\Theta_r$  is the reference concentration of the scalar  $\Theta$ .

After substitution of the variable scaling, the equations (3.18), (3.37) and (3.39) become:

$$\frac{\partial U_i^*}{\partial x_i^*} = 0 \quad (3.58a)$$

$$\frac{\overline{D}U_i^*}{Dt^*} = -\frac{\partial P^*}{\partial x_i^*} + \frac{\partial}{\partial x_j^*} \left[ \frac{1}{Re_{eff}} \left( \frac{\partial U_i^*}{\partial x_j^*} + \frac{\partial U_j^*}{\partial x_i^*} \right) \right] - \frac{1}{Fr^2} \Theta^* g_i^* \quad (3.58b)$$

$$\frac{\overline{D}\Theta^*}{Dt^*} = \frac{\partial}{\partial x_j^*} \left( \frac{1}{Sc_{eff} Re_{eff}} \frac{\partial \Theta^*}{\partial x_j^*} \right) + S_\theta^* \quad (3.58c)$$

where,  $Re_{eff}$ , defined as

$$Re_{eff} = \frac{U_c L_c}{\nu_{eff}} \quad (3.59)$$

is the effective Reynolds number. It is similar to the Reynolds number defined above, but in this case the effective viscosity is used instead the molecular viscosity. Based on the turbulence model for stratified flow (equations 3.52 and 3.53), the effective Reynolds numbers can be written as

$$Re_{eff}^{-1} = \begin{cases} 2l^{*2} \sqrt{S_{ij}^* S_{ij}^*} (1 - Ri/Ri_c)^2 + Re^{-1} & 0 \leq Ri \leq Ri_c \\ Re^{-1} & Ri \geq Ri_c \end{cases} \quad (3.60)$$

for stable stratification ( $Ri > 0$ ) and

$$Re_{eff}^{-1} = 2l^{*2} \sqrt{S_{ij}^* S_{ij}^*} (1 - Ri)^{1/2} + Re^{-1} \quad (3.61)$$

for unstable stratification, where  $l^*$  and  $S^*$  are defined as

$$S_{ij}^* = S_{ij} \frac{L_c}{U_c} \quad l^* = \frac{l}{L_c} \quad (3.62)$$

and  $Re$  is defined as

$$Re = \frac{U_c L_c}{\nu} \quad (3.63)$$

where  $\nu$  is the molecular viscosity.

Other dimensionless number  $Fr$ , which appears in the equation (3.58b), is the Froude number, and is defined as

$$Fr = \frac{U_c}{\sqrt{g' L_c}} \quad (3.64)$$

This dimensionless number gives ratio of inertia forces and gravity forces. The  $g'$  is called reduced gravity and is defined as

$$g' = \beta \Theta_r g \quad (3.65)$$

The reduced gravity is also commonly defined as

$$g' = \frac{\rho - \rho_r}{\rho_r} g \quad (3.66)$$

which can be obtained using the equation (3.13).

The Schmidt number that gives rate of the momentum diffusivity and scalar diffusivity, is defined as

$$Sc_{eff} = \frac{\nu_{eff}}{\alpha_{eff}} \quad (3.67)$$

For convenience, the asterisks and the subscript term  $._{eff}$  are dropped from the non-dimensional equations (equations 3.58).

# CHAPTER 4

---

## DISCRETIZATION METHOD

---

This chapter describes the discretization methods used to obtain the flow solution. The variational approach (weak form) of the RANS and mean scalar transport equations system are shown. Then, the Galerkin semi-discrete method for spatial discretization and semi-Lagrangian method for the convective terms discretization are presented.

### 4.1 Variational approaches

The fluid flow solution is given by the functions  $U_i (i = 1, 2, 3)$ ,  $P$  and  $\Theta$  defined in  $(\Omega, t) \subset \mathbb{R}^3 \times \mathbb{R}^+$ , which satisfy the following differential equations system

$$\frac{\partial U_i}{\partial x_i} = 0 \quad (4.1a)$$

$$\frac{\overline{D}U_i}{Dt} = -\frac{\partial P}{\partial x_i} + \frac{\partial}{\partial x_j} \left[ \frac{1}{Re} \left( \frac{\partial U_i}{\partial x_j} + \frac{\partial U_j}{\partial x_i} \right) \right] - \frac{1}{Fr^2} \Theta g_i \quad (4.1b)$$

$$\frac{\overline{D}\Theta}{Dt} = \frac{\partial}{\partial x_j} \left( \frac{1}{Sc Re} \frac{\partial \Theta}{\partial x_j} \right) + S_\theta \quad (4.1c)$$

restricted to the following boundary conditions

$$U_i = U_{\Gamma_i} \quad \text{on} \quad \Gamma_i; \quad \frac{\partial U_i}{\partial x_n} = U_{\Gamma_i^c} \quad \text{on} \quad \Gamma_i^c \quad (i = 1, 2, 3) \quad (4.2a)$$

$$P = P_{\Gamma_p} \quad \text{on} \quad \Gamma_p; \quad \frac{\partial P}{\partial x_n} = P_{\Gamma_p^c} \quad \text{on} \quad \Gamma_p^c \quad (4.2b)$$

$$\Theta = \Theta_{\Gamma_\theta} \quad \text{on} \quad \Gamma_\theta; \quad \frac{\partial \Theta}{\partial x_n} = \Theta_{\Gamma_\theta^c} \quad \text{on} \quad \Gamma_\theta^c \quad (4.2c)$$

and initial conditions

$$U_i = U_{it_0} \quad P = P_{t_0} \quad \Theta = \Theta_{t_0} \quad \text{in} \quad \Omega \quad \text{at} \quad t = t_0 \quad (4.3)$$

where  $\Omega$  is the domain of the differential equations system, and  $\Gamma = \Gamma_\xi \oplus \Gamma_\xi^c$  ( $\Gamma = \Gamma_\xi \cup \Gamma_\xi^c$  and  $\Gamma_\xi \cap \Gamma_\xi^c = \emptyset$ ) are the boundary of  $\Omega$ , *i.e.*,  $\Gamma = \partial\Omega$ . The Dirichlet boundary condition is prescribed on the part of the boundary denoted by  $\Gamma_\xi$  and the Neumann boundary condition is prescribed on the other part denoted by  $\Gamma_\xi^c$ .

Let  $\mathcal{L}^2(\Omega)$  be the space of functions defined in  $\Omega$  that are square integrable over  $\Omega$ , *i.e.*,

$$\mathcal{L}^2(\Omega) = \left\{ v : \Omega \rightarrow \mathbb{R}; \int_{\Omega} v^2 d\Omega < \infty \right\} \quad (4.4)$$

the Sobolev space of degree one  $\mathcal{H}^1(\Omega)$  is defined as the space of functions defined over  $\Omega$  such that the function and its first partial derivatives, in the weak sense, are in  $\mathcal{L}^2(\Omega)$ . That is,

$$\mathcal{H}^1(\Omega) = \left\{ v : \Omega \rightarrow \mathbb{R}; v, \frac{\partial v}{\partial x_i} \in \mathcal{L}^2(\Omega), i = 1, 2, 3 \right\} \quad (4.5)$$

The subspace  $\mathcal{H}_\xi^1(\Omega)$  is defined as the subset of  $\mathcal{H}^1(\Omega)$  that takes the value of the function  $\xi$  on the boundary  $\Gamma_\xi$  where Dirichlet boundary conditions are imposed, *i.e.*,

$$\mathcal{H}_\xi^1(\Omega) = \{ v \in \mathcal{H}^1(\Omega); v = \xi \text{ in } \Gamma_\xi \} \quad (4.6)$$

and, particularly for  $\xi = 0$  (called homogeneous boundary condition),

$$\mathcal{H}_0^1(\Omega) = \{ v \in \mathcal{H}^1(\Omega); v = 0 \text{ in } \Gamma_\xi \} \quad (4.7)$$

Let  $w_i (i = 1, 2, 3)$ ,  $w_p$  and  $w_\theta \in \mathcal{H}_0^1$ , called weighting functions, and the residual func-



tions of the equations (4.1):

$$\mathcal{R}_1 = \frac{\partial U_i}{\partial x_i} \quad (4.8a)$$

$$\mathcal{R}_{2i} = \frac{\overline{D}U_i}{Dt} + \frac{\partial P}{\partial x_i} - \frac{\partial}{\partial x_j} \left[ \frac{1}{Re} \left( \frac{\partial U_i}{\partial x_j} + \frac{\partial U_j}{\partial x_i} \right) \right] + \frac{1}{Fr^2} \Theta g_i \quad (4.8b)$$

$$\mathcal{R}_3 = \frac{\overline{D}\Theta}{Dt} - \frac{\partial}{\partial x_j} \left( \frac{1}{Sc Re} \frac{\partial \Theta}{\partial x_j} \right) - S_\theta \quad (4.8c)$$

The weighted residual form is obtained by multiplying the residual functions (equations 4.8) by the weighting functions, integrating over the domain ( $\Omega$ ), and equating the integrals to zero, as follows

$$\int_{\Omega} w_p \mathcal{R}_1 d\Omega = \int_{\Omega} w_p \frac{\partial U_i}{\partial x_i} d\Omega = 0 \quad (4.9a)$$

$$\int_{\Omega} w_i \mathcal{R}_{2i} d\Omega = \int_{\Omega} w_i \left\{ \frac{\overline{D}U_i}{Dt} + \frac{\partial P}{\partial x_i} - \frac{\partial}{\partial x_j} \left[ \frac{1}{Re} \left( \frac{\partial U_i}{\partial x_j} + \frac{\partial U_j}{\partial x_i} \right) \right] + \frac{1}{Fr^2} \Theta g_i \right\} d\Omega = 0 \quad (4.9b)$$

$$\int_{\Omega} w_\theta \mathcal{R}_3 d\Omega = \int_{\Omega} w_\theta \left\{ \frac{\overline{D}\Theta}{Dt} - \frac{\partial}{\partial x_j} \left( \frac{1}{Sc Re} \frac{\partial \Theta}{\partial x_j} \right) - S_\theta \right\} d\Omega = 0 \quad (4.9c)$$

The Green's identities, can be applied in the following form:

$$\begin{aligned} \int_{\Omega} w_i \frac{\partial}{\partial x_j} \left[ \frac{1}{Re} \left( \frac{\partial U_i}{\partial x_j} + \frac{\partial U_j}{\partial x_i} \right) \right] d\Omega &= \int_{\Gamma_i} \frac{1}{Re} w_i \left( \frac{\partial U_i}{\partial x_j} + \frac{\partial U_j}{\partial x_i} \right) n_j d\Gamma \\ &\quad - \int_{\Omega} \frac{1}{Re} \frac{\partial w_i}{\partial x_j} \left( \frac{\partial U_i}{\partial x_j} + \frac{\partial U_j}{\partial x_i} \right) d\Omega \end{aligned} \quad (4.10a)$$

$$\int_{\Omega} w_\theta \frac{\partial}{\partial x_j} \left( \frac{1}{Sc Re} \frac{\partial \Theta}{\partial x_j} \right) d\Omega = \int_{\Gamma_\theta} \frac{1}{Sc Re} w_\theta \frac{\partial \Theta}{\partial x_j} n_j d\Gamma - \int_{\Omega} \frac{1}{Sc Re} \frac{\partial w_\theta}{\partial x_j} \frac{\partial \Theta}{\partial x_j} d\Omega \quad (4.10b)$$

where,  $n_i (i = 1, 2, 3)$  is the component of the vector normal to the boundary in each point. Notice that the integrals along the boundary of  $\Omega$  can be separated into two integrals according to the boundary conditions type (Dirichlet or Neumann). The integrals along the boundary  $\Gamma_\xi (\xi = U_i, P, \Theta)$  are zero, because the weighting functions  $w_p, w_i (i = 1, 2, 3)$  and  $w_\theta$  belong to the subspace  $\mathcal{H}_0^1$ .

Substituting the equations (4.10) into the equations (4.9)

$$\int_{\Omega} w_p \frac{\partial U_i}{\partial x_i} d\Omega = 0 \quad (4.11a)$$

$$\begin{aligned} \int_{\Omega} w_i \frac{\overline{DU}_i}{Dt} d\Omega + \int_{\Omega} w_i \frac{\partial P}{\partial x_i} \Omega + \int_{\Omega} \frac{1}{Re} \frac{\partial w_i}{\partial x_j} \left( \frac{\partial U_i}{\partial x_j} + \frac{\partial U_j}{\partial x_i} \right) d\Omega + \int_{\Omega} w_i \frac{1}{Fr^2} \Theta g_i d\Omega \\ = \int_{\Gamma_i^c} \frac{1}{Re} w_i \left( \frac{\partial U_i}{\partial x_j} + \frac{\partial U_j}{\partial x_i} \right) n_j d\Gamma \end{aligned} \quad (4.11b)$$

$$\int_{\Omega} w_{\theta} \frac{\overline{D\Theta}}{Dt} d\Omega + \int_{\Omega} \frac{1}{Sc Re} \frac{\partial w_{\theta}}{\partial x_j} \frac{\partial \Theta}{\partial x_j} d\Omega = \int_{\Gamma_{\theta}^c} \frac{1}{Sc Re} w_{\theta} \frac{\partial \Theta}{\partial x_j} n_j d\Gamma + \int_{\Omega} w_{\theta} S_{\theta} d\Omega \quad (4.11c)$$

A solution of the system (4.1) is also a solution of the system (4.11). However, while the solution of the system (4.1) has the restriction that it must have second order derivatives in  $\Omega$ , the solution of the system (4.11) only requires to be differentiable to first order. This means that the differentiability requirements of the solution have been relaxed. For this reason, the system (4.11) is called the weak form or variational form of the system (4.1).

The variational approach of the differential equations system (4.1) reads: Seek functions  $U_i \in \mathcal{H}_{U_i}^1 \times \mathbb{R}^+$  ( $i = 1, 2, 3$ ),  $P \in \mathcal{L}^2 \times \mathbb{R}^+$  and  $\Theta \in \mathcal{H}_{\Theta}^1 \times \mathbb{R}^+$ , such that verify the weak form (equation 4.11) of the problem for all  $w_p, w_i$  ( $i = 1, 2, 3$ ) and  $w_{\theta} \in \mathcal{H}_0^1$ .

## 4.2 Element of mesh

To solve numerically the fluid flow problem, the domain where the functions are defined is discretized. This means, instead of the functions defined in infinite dimensional space, an approximate solution is sought in a finite dimensional function space. Seek an approximate solution of a variational problem in a finite dimensional space means write the solution as a linear combination of elements of the basis (with undetermined coefficient) that generates the space and then solve the algebraic linear equations system obtained. The choice of the basis for the space defines the degree of difficulty in solving the linear system. For this reason, an appropriate choice of basis is important for the performance of the numerical simulation.

The FEM provides techniques for constructing appropriate basis, using functions, which are smooth enough, defined piecewise over small subregion of the domain. Smooth enough functions are obtained by belonging to the Sobolev space of degree one, which is the search space for the solution of the variational form. Piecewise functions with compact support are defined generating a sparse linear system. This is desirable to reduce the computational cost for solving the linear system. Moreover, the functions of the basis are such that the coefficients of the linear combination are the values of the solution in these nodes. This allows the imposition

of Dirichlet boundary conditions on the variables on an easy way.

In FEM, the domain is divided into subregions in which the functions for the basis of the solution space are defined. These subregions are called elements. The function of the basis is called interpolating function or shape function. The element shape should be such that the interpolation functions defined in it satisfy the requirements mentioned in the paragraph above.

In principle, it is possible to employ the same interpolation space for all variables. But in incompressible problems such as the one being treated, there are numerical difficulties due to the constraint on the velocity field to satisfy the divergent free condition. To avoid this problem, the Ladyzhenskaya, Babuska and Brezzi (LBB) condition, which says that the velocity and pressure spaces must be consistent, must be satisfied. In the figure 4.1 shows the MINI element which belongs to the Taylor-Hood element family, which fulfills the LBB condition.

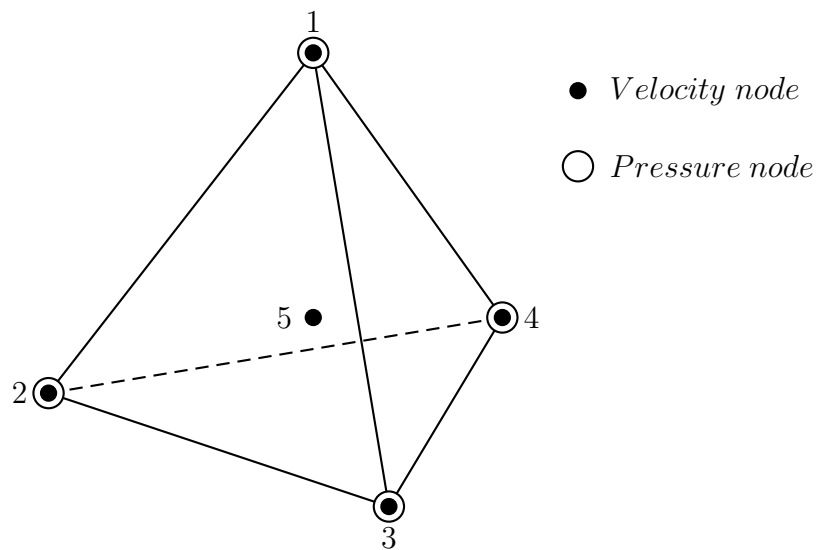


Figure 4.1: Mini tetrahedron element.

The vertices nodes of the tetrahedron are used to calculate the pressure and the concentration field and the centroid node (also called bubble node) combined with the vertices nodes to calculate the velocity.

#### 4.2.1 Volume coordinates

A new set of coordinates,  $L_i (i = 1, 2, 3, 4)$  for a tetrahedron is defined. Consider the tetrahedron with vertices enumerated as in figure 4.1. Let  $x_{ij}$  be the  $j$ -component ( $j = 1, 2, 3$ ) of the vertices  $i (i = 1, 2, 3, 4)$ . The coordinate of a point  $Q(x_j)$ , interior to tetrahedron, related

to the new set of coordinates is given as follows,

$$x_j = L_i x_{ij} \quad (4.12a)$$

$$1 = \sum_{i=1}^4 L_i \quad (4.12b)$$

Solving equations 4.12 gives

$$L_i = \frac{V_i}{V} \quad (i = 1, 2, 3, 4) \quad (4.13)$$

where  $V$  is the volume of the tetrahedron 1234, and  $V_i$  is the volume of the tetrahedron  $Qjkl$  with  $i \neq j$ ,  $i \neq k$ , and  $i \neq l$ . The set of coordinates  $L_i (i = 1, 2, 3, 4)$  are called volume coordinates.

#### 4.2.2 Shape function for tetrahedra

Let  $ne$  be the number of elements that form the domain and  $\Omega^e$  the subregion of this domain defined for the element  $e (e = 1, \dots, ne)$ . Henceforth, a superscript  $e$  is placed on the variable to indicate that this variable is defined locally in the element  $e$ . The domain  $\Omega$  is

$$\Omega \approx \bigcup_{e=1}^{ne} \Omega^e \quad (4.14)$$

And further, let  $n$  be the number of nodes of the discrete domain. It is convenient to define the interpolation function  $\varphi_m$  of the node  $m (m = 1, \dots, n)$ , such that, at node  $n$   $\varphi_m(x_n) = \delta_{mn}$ .

The piecewise shape functions are obtained by joining the interpolation function restricted to the tetrahedron domain  $\Omega^e$ , *i.e.*

$$\varphi_s^e = \varphi_m|_{\Omega^e} \quad (4.15)$$

where,  $s$  is the local enumeration for the node  $x_m$  (that is,  $x_s^e = x_m$ ). The restricted interpolation functions are given by

$$\varphi_i^e = L_i \quad (i = 1, 2, 3, 4) \quad (4.16)$$

for the linear tetrahedron, *i.e.*, using only the vertices of the tetrahedron, and

$$\varphi_i^e = L_i - 64L_1L_2L_3L_4 \quad (i = 1, 2, 3, 4) \quad (4.17a)$$

$$\varphi_5^e = 256L_1L_2L_3L_4 \quad (4.17b)$$

for the mini tetrahedron, *i.e.*, using five nodes (the vertices with the bubble node).

Let  $\mathcal{H}^{(n)}$  be the Sobolev space of degree one with dimension  $n$ . The interpolation functions  $\varphi_m (m = 1, \dots, n)$  form a basis of the space  $\mathcal{H}^{(n)}$ .

### 4.3 Galerkin method

The unknown functions of the variational form the equations (4.11) belong to  $\mathcal{H}^1$ . However, these functions can be separated as

$$U_i = U_{\Gamma_i} + \widehat{U}_i \quad (i = 1, 2, 3) \quad (4.18a)$$

$$P = P_{\Gamma_p} + \widehat{P} \quad (4.18b)$$

$$\Theta = \Theta_{\Gamma_\theta} + \widehat{\Theta} \quad (4.18c)$$

where,  $U_{\Gamma_i} \in \mathcal{H}_{\Gamma_i}^1 (i = 1, 2, 3)$ ,  $P_{\Gamma_p} \in \mathcal{H}_{\Gamma_p}^1$ ,  $\Theta_{\Gamma_\theta} \in \mathcal{H}_{\Gamma_\theta}^1$ , and  $\widehat{U}_i (i = 1, 2, 3)$ ,  $\widehat{P}$  and  $\widehat{\Theta} \in \mathcal{H}_0^1$ . That is, each variable is separated into a function that satisfies the specified Dirichlet boundary conditions and another with homogeneous boundary conditions. The first is known, and the second belongs to  $\mathcal{H}_0^1$ .

The Galerkin method is based on making the projection of the residue on the search space of solution being zero. This is achieved by making the interpolation and weighting functions belong to the same space,  $\mathcal{H}_0^1$ .

Let  $nu$  be the number of nodes for velocity,  $np$  be the number of nodes for pressure and  $n\theta$  be the number of nodes for the scalar concentration. The spaces in which the approximate solutions for velocity, pressure and scalar are  $\mathcal{H}^{(nu)}$ ,  $\mathcal{H}^{(np)}$  and  $\mathcal{H}^{(n\theta)}$ , respectively. The basis for these spaces are  $\varphi_m (m = 1, \dots, nu) \in \mathcal{H}^{(nu)}$ ,  $\chi_m (m = 1, \dots, np) \in \mathcal{H}^{(np)}$ ,  $\psi_m (m = 1, \dots, n\theta) \in \mathcal{H}^{(n\theta)}$ . The functions of the equations (4.18) are written as linear combination of

the basis functions, as follows

$$U_i(x_j, t) \approx \sum_{m \in \eta_u - \eta_{\Gamma_i}} \varphi_m(x_j) U_{im}(t) + \sum_{m \in \eta_{\Gamma_i}} \varphi_m(x_j) U_{\Gamma_i m} \quad (i = 1, 2, 3) \quad (4.19a)$$

$$P(x_j, t) \approx \sum_{m \in \eta_p - \eta_{\Gamma_p}} \chi_m(x_j) P_m(t) + \sum_{m \in \eta_{\Gamma_p}} \chi_m(x_j) P_{\Gamma_p m} \quad (4.19b)$$

$$\Theta(x_j, t) \approx \sum_{m \in \eta_\theta - \eta_{\Gamma_\theta}} \psi_m(x_j) \Theta_m(t) + \sum_{m \in \eta_{\Gamma_\theta}} \psi_m(x_j) \Theta_{\Gamma_\theta m} \quad (4.19c)$$

where  $\eta_u = \{1, \dots, nu\}$ ,  $\eta_p = \{1, \dots, np\}$  and  $\eta_\theta = \{1, \dots, n\theta\}$  are the sets of node number for velocity, pressure and scalar concentration respectively,  $\eta_{\Gamma_i} = \{m \in \eta_u; x_m \in \Gamma_i\}$ ,  $\eta_{\Gamma_p} = \{m \in \eta_p; x_m \in \Gamma_p\}$ , and  $\eta_{\Gamma_\theta} = \{m \in \eta_\theta; x_m \in \Gamma_\theta\}$  the sets of node at which Dirichlet boundary conditions are prescribed,  $U_{im}(t)$  ( $m \in \eta_u - \eta_{\Gamma_i}$ ),  $P_m(t)$  ( $m \in \eta_p - \eta_{\Gamma_p}$ ) and  $\Theta_m(t)$  ( $m \in \eta_\theta - \eta_{\Gamma_\theta}$ ) are undetermined coefficients, and  $U_{\Gamma_i m}$  ( $m \in \eta_{\Gamma_i}$ ),  $P_{\Gamma_p m}$  ( $m \in \eta_{\Gamma_p}$ ) and  $\Theta_{\Gamma_\theta m}$  ( $m \in \eta_{\Gamma_\theta}$ ) are the nodes values on the Dirichlet boundary conditions. The unknown coefficients are continuous in time. This means that the approximate solution is time dependent. For this reason, this approximation is called semi-discrete approach.

The subspace of  $\mathcal{H}^{(n)}$  with zero value at points of Dirichlet boundary conditions, is denoted by  $\mathcal{H}_0^{(n)}$ . Notice that  $\varphi_m$  ( $m \in \eta_u - \eta_{\Gamma_i}$ ),  $\chi_m$  ( $m \in \eta_p - \eta_{\Gamma_p}$ ) and  $\psi_m$  ( $m \in \eta_\theta - \eta_{\Gamma_\theta}$ ) form the basis of  $\mathcal{H}_0^{(nu)}$ ,  $\mathcal{H}_0^{(np)}$  and  $\mathcal{H}_0^{(n\theta)}$  respectively. The weighting functions belong to  $\mathcal{H}_0^{(nu)}$ ,  $\mathcal{H}_0^{(np)}$  and  $\mathcal{H}_0^{(n\theta)}$ . That is,

$$w_i = \sum_{m \in \eta_u - \eta_{\Gamma_i}} a_{im} \varphi_m(x_j) \quad (i = 1, 2, 3) \quad (4.20a)$$

$$w_p = \sum_{m \in \eta_p - \eta_{\Gamma_p}} b_m \chi_m(x_j) \quad (4.20b)$$

$$w_\theta = \sum_{m \in \eta_\theta - \eta_{\Gamma_\theta}} c_m \psi_m(x_j) \quad (4.20c)$$

where  $a_{im}$  ( $m \in \eta_u - \eta_{\Gamma_i}$ ,  $i = 1, 2, 3$ ),  $b_m$  ( $m \in \eta_p - \eta_{\Gamma_p}$ ), and  $c_m$  ( $m \in \eta_\theta - \eta_{\Gamma_\theta}$ ) are non-zero coefficients.

The semi-discret form of the variational equations is obtained substituting the equations (4.19) and (4.20) into the equations (4.11). Performing summation of the integrals restricted to

the elements domain, the following equations are obtained

$$\sum_e \int_{\Omega^e} \sum_{m,n \in e} \chi_m^e \frac{\partial \varphi_n^e}{\partial x_i} U_{in} d\Omega = 0 \quad (4.21a)$$

$$\begin{aligned} & \sum_e \int_{\Omega^e} \sum_{m,n \in e} \varphi_m^e \varphi_n^e \frac{\overline{D}U_{in}}{Dt} d\Omega + \sum_e \int_{\Omega^e} \sum_{m,n \in e} \varphi_m^e \frac{\partial \chi_n^e}{\partial x_i} P_n d\Omega \\ & + \sum_e \frac{1}{Re^e} \int_{\Omega^e} \sum_{m,n \in e} \frac{\partial \varphi_m^e}{\partial x_j} \left( \frac{\partial \varphi_n^e}{\partial x_j} U_{in} + \frac{\partial \varphi_n^e}{\partial x_i} U_{jn} \right) d\Omega + \sum_e \int_{\Omega^e} \sum_{m,n \in e} \varphi_m^e \frac{1}{Fr^2} \psi_n^e \Theta_n g_i d\Omega \\ & = \sum_e \frac{1}{Re^e} \int_{\Gamma_i^e} \sum_{m,n \in e} \varphi_m^e \left( \frac{\partial \varphi_n^e}{\partial x_j} U_{in} + \frac{\partial \varphi_n^e}{\partial x_i} U_{jn} \right) n_j d\Gamma \end{aligned} \quad (4.21b)$$

$$\begin{aligned} & \sum_e \int_{\Omega^e} \sum_{m,n \in e} \psi_m^e \psi_n^e \frac{\overline{D}\Theta}{Dt} d\Omega + \sum_e \frac{1}{Sc^e Re^e} \int_{\Omega^e} \sum_{m,n \in e} \frac{\partial \psi_m^e}{\partial x_j} \frac{\partial \psi_n^e}{\partial x_j} \Theta_n d\Omega \\ & = \sum_e \frac{1}{Sc^e Re^e} \int_{\Gamma_\theta^e} \sum_{m,n \in e} \psi_m^e \frac{\partial \psi_n^e}{\partial x_j} \Theta_n n_j d\Gamma + \sum_e \int_{\Omega^e} \sum_{m,n \in e} \psi_m^e S_\theta d\Omega \end{aligned} \quad (4.21c)$$

These equations can be represented as a system of ordinary differential equations

$$D_i \tilde{u}_i = 0 \quad (4.22a)$$

$$M_\rho \dot{\tilde{u}}_i + G_i \tilde{p} + K_{\rho jj} \tilde{u}_i + K_{\rho ji} \tilde{u}_j + \frac{1}{Fr^2} g_i F \tilde{\theta} = bc n_{2i} \quad (i = 1, 2, 3) \quad (4.22b)$$

$$M_\theta \dot{\tilde{\theta}} + K_{\theta i} \tilde{\theta} = bc n_3 \quad (4.22c)$$

The unknown variables are

$$\tilde{u}_i = [ \tilde{u}_{ia} ]_{nu \times 1}, \quad \tilde{u}_{ia} = U_{ia}, \quad 1 \leq a \leq nu, \quad i = 1, 2, 3 \quad (4.23a)$$

$$\tilde{p} = [ \tilde{p}_a ]_{np \times 1}, \quad \tilde{p}_a = P_a, \quad 1 \leq a \leq np \quad (4.23b)$$

$$\tilde{\theta} = [ \tilde{\theta}_a ]_{n\theta \times 1}, \quad \tilde{\theta}_a = \Theta_a, \quad 1 \leq a \leq n\theta \quad (4.23c)$$

where the values with subscript  $a$  are nodal values. The substantial derivatives of the velocity and the scalar concentration are represented as  $\dot{\tilde{u}}_i (i = 1, 2, 3)$  and  $\dot{\tilde{\theta}}$  given by

$$\dot{\tilde{u}}_i = [ \dot{\tilde{u}}_{ia} ]_{nu \times 1}, \quad \dot{\tilde{u}}_{ia} = \frac{\overline{D}U_{ia}}{Dt}, \quad 1 \leq a \leq nu, \quad i = 1, 2, 3 \quad (4.24a)$$

$$\dot{\tilde{\theta}} = [ \dot{\tilde{\theta}}_a ]_{n\theta \times 1}, \quad \dot{\tilde{\theta}}_a = \frac{\overline{D}\Theta_a}{Dt}, \quad 1 \leq a \leq n\theta \quad (4.24b)$$

The matrices are

$$D_i = [ D_{i ab} ]_{np \times nu}, \quad 1 \leq a \leq np, \quad 1 \leq b \leq nu, \quad i = 1, 2, 3 \quad (4.25a)$$

$$M_\rho = [ M_{\rho ab} ]_{nu \times nu}, \quad 1 \leq a \leq nu, \quad 1 \leq b \leq nu, \quad (4.25b)$$

$$G_i = [ G_{i ab} ]_{nu \times np}, \quad 1 \leq a \leq nu, \quad 1 \leq b \leq np, \quad i = 1, 2, 3 \quad (4.25c)$$

$$K_{\rho ij} = [ K_{\rho ij ab} ]_{nu \times nu}, \quad 1 \leq a \leq nu, \quad 1 \leq b \leq nu, \quad i, j = 1, 2, 3 \quad (4.25d)$$

$$F = [ F_{ab} ]_{nu \times n\theta}, \quad 1 \leq a \leq nu, \quad 1 \leq b \leq n\theta, \quad (4.25e)$$

$$M_\theta = [ M_{\theta ab} ]_{n\theta \times n\theta}, \quad 1 \leq a \leq n\theta, \quad 1 \leq b \leq n\theta, \quad (4.25f)$$

$$K_{\theta i} = [ K_{\theta i ab} ]_{n\theta \times n\theta}, \quad 1 \leq a \leq n\theta, \quad 1 \leq b \leq n\theta, \quad i = 1, 2, 3 \quad (4.25g)$$

To assemble these matrices an auxiliary operator that relates the global node numbers with the local node numbers is used. This operator is called location matrix, and it is defined as  $LM : (m, n, e) \mapsto (a, b)$ , where  $m, n$  are the local node numeration,  $e$  is the element identifier, and  $a, b$  global node enumeration. The following relations show how the matrices of the equations (4.25) are mounted

$$D_{i ab} \leftarrow D_{i ab} + D_{i mn}^e; \quad D_{i mn}^e = \int_{\Omega^e} \chi_m^e \frac{\partial \varphi_n^e}{\partial x_i} d\Omega \quad (i = 1, 2, 3) \quad (4.26a)$$

$$M_{\rho ab} \leftarrow M_{\rho ab} + M_{\rho mn}^e; \quad M_{\rho mn}^e = \int_{\Omega^e} \varphi_m^e \varphi_n^e d\Omega \quad (4.26b)$$

$$G_{i ab} \leftarrow G_{i ab} + G_{i mn}^e; \quad G_{i mn}^e = \int_{\Omega^e} \varphi_m^e \frac{\partial \chi_n^e}{\partial x_i} d\Omega \quad (i = 1, 2, 3) \quad (4.26c)$$

$$K_{\rho ij ab} \leftarrow K_{\rho ij ab} + K_{\rho ij mn}^e; \quad K_{\rho ij mn}^e = \frac{1}{Re^e} \int_{\Omega^e} \frac{\partial \varphi_m^e}{\partial x_i} \frac{\partial \varphi_n^e}{\partial x_j} d\Omega \quad (i, j = 1, 2, 3) \quad (4.26d)$$

$$F_{ab} \leftarrow F_{ab} + F_{mn}^e; \quad F_{mn}^e = \int_{\Omega^e} \varphi_m^e \psi_n^e d\Omega \quad (4.26e)$$

$$M_{\theta ab} \leftarrow M_{\theta ab} + M_{\theta mn}^e; \quad M_{\theta mn}^e = \int_{\Omega^e} \psi_m^e \psi_n^e d\Omega \quad (4.26f)$$

$$K_{\theta i ab} \leftarrow K_{\theta i ab} + K_{\theta i mn}^e; \quad K_{\theta i mn}^e = \frac{1}{Sc^e Re^e} \int_{\Omega^e} \frac{\partial \psi_m^e}{\partial x_i} \frac{\partial \psi_n^e}{\partial x_i} d\Omega \quad (i, j = 1, 2, 3) \quad (4.26g)$$

The equation (4.22) is written in matrix form as

$$\mathbf{D}\tilde{\mathbf{u}} = \mathbf{0} \quad (4.27a)$$

$$\mathbf{M}_\rho \dot{\tilde{\mathbf{u}}} + \mathbf{G}\tilde{\mathbf{p}} + \mathbf{K}_\rho \tilde{\mathbf{u}} + \frac{1}{Fr^2} \mathbf{g}\mathbf{F}\tilde{\theta} = \mathbf{bcn}_2 \quad (4.27b)$$



$$\mathbf{M}_\theta \dot{\tilde{\theta}} + \mathbf{K}_\theta \tilde{\theta} = \mathbf{bcn}_3 \quad (4.27c)$$

where,  $\mathbf{D}$  is the divergence matrix,  $\mathbf{G}$  is the gradient matrix,  $\mathbf{M}_\rho$  is the mass matrix,  $\mathbf{K}_\rho$  is the momentum diffusion matrix,  $\mathbf{M}_\theta$  is the scalar mass matrix,  $\mathbf{K}_\theta$  is the scalar diffusion matrix. These matrices are given by

$$\mathbf{D} = \begin{bmatrix} D_1 & D_2 & D_3 \end{bmatrix}_{np \times 3nu} \quad (4.28a)$$

$$\mathbf{G} = \begin{bmatrix} G_1 \\ G_2 \\ G_3 \end{bmatrix}_{3nu \times np} \quad (4.28b)$$

$$\mathbf{M}_\rho = \begin{bmatrix} M_\rho & 0 & 0 \\ 0 & M_\rho & 0 \\ 0 & 0 & M_\rho \end{bmatrix}_{3nu \times 3nu} \quad (4.28c)$$

$$\mathbf{K}_\rho = \begin{bmatrix} 2K_{\rho 11} + K_{\rho 22} + K_{\rho 33} & K_{\rho 12} & K_{\rho 13} \\ K_{\rho 21} & K_{\rho 11} + 2K_{\rho 22} + K_{\rho 33} & K_{\rho 23} \\ K_{\rho 31} & K_{\rho 32} & K_{\rho 11} + K_{\rho 22} + 2K_{\rho 33} \end{bmatrix}_{3nu \times 3nu} \quad (4.28d)$$

$$\mathbf{M}_\theta = \begin{bmatrix} M_\theta \end{bmatrix}_{n\theta \times n\theta} \quad (4.28e)$$

$$\mathbf{K}_\theta = \begin{bmatrix} K_{\theta 1} + K_{\theta 2} + K_{\theta 3} \end{bmatrix}_{n\theta \times n\theta} \quad (4.28f)$$

and the unknown variables vector are given by

$$\tilde{\mathbf{p}} = \begin{bmatrix} p \end{bmatrix}_{np \times 1} \quad (4.29a)$$

$$\tilde{\mathbf{u}} = \begin{bmatrix} \tilde{u}_1 \\ \tilde{u}_2 \\ \tilde{u}_3 \end{bmatrix}_{3nu \times 1} \quad (4.29b)$$

$$\tilde{\theta} = \begin{bmatrix} \tilde{\theta} \end{bmatrix}_{n\theta \times 1} \quad (4.29c)$$

#### 4.4 Semi-Lagrangian method

The ordinary differential equations system (equations 4.27) is solved employing the semi-Lagrangian method for time discretization. Because of the larger allowable time step, the semi-Lagrangian technique contributes to a significant enhancement of the efficiency of the semi-implicit integration scheme (Robert et al., 1984). Using a function  $\phi$ , the substantive derivative of this function at the point  $x_i$  can be discretized using a first order scheme as

$$\frac{\overline{D}\phi}{Dt} = \frac{\phi_m^{n+1} - \phi_d^n}{\Delta t} \quad (4.30)$$

where,  $\phi_m^{n+1} = \phi(x_m, t^{n+1})$  is the image of  $\phi$  at the point  $x_m$  and the time step  $n + 1$  and  $\phi_d^n = \phi(x_d, t^n)$  is the image of  $\phi$  at the point  $x_d$  and the time step  $n$ , obtained by interpolating the solution on the mesh nodes at time step  $n$ . The  $i$ -component of the position  $x_d$  is obtained using the expression

$$x_{id} = x_{im} - u_i \Delta t \quad (i = 1, 2, 3) \quad (4.31)$$

where  $u_i = u_i(x_m, t^n)$  is the velocity vector at the point  $x_m$  and time step  $n$ .

Equations (4.27), with time discretization read

$$\mathbf{D}\tilde{\mathbf{u}}^{n+1} = \mathbf{0} \quad (4.32a)$$

$$\mathbf{M}_\rho \left( \frac{\tilde{\mathbf{u}}^{n+1} - \tilde{\mathbf{u}}_d^n}{\Delta t} \right) + \mathbf{G}\tilde{\mathbf{p}}^{n+1} + \mathbf{K}_\rho (\lambda \tilde{\mathbf{u}}^{n+1} + (1 - \lambda)\tilde{\mathbf{u}}_d^n) + \frac{1}{Fr^2} \mathbf{gF}\tilde{\theta}_d^n = \mathbf{bcn}_2 \quad (4.32b)$$

$$\mathbf{M}_\theta \left( \frac{\tilde{\theta}^{n+1} - \tilde{\theta}_d^n}{\Delta t} \right) + \mathbf{K}_\theta (\lambda \tilde{\theta}^{n+1} + (1 - \lambda)\tilde{\theta}_d^n) = \mathbf{bcn}_3 \quad (4.32c)$$

where,  $\lambda$  is a parameter to obtain different methods of discretization in time. For  $\lambda = 0$  results a explicit discretization,  $\lambda = 1$  results a semi-implicit discretization and  $\lambda = \frac{1}{2}$  results the Crank-Nicolson method.

#### 4.5 Projection method

The linear equations system (4.32) is solved employing the block LU approximate factorization. The linear system (4.32) is rewritten as

$$\mathbf{D}\tilde{\mathbf{u}}^{n+1} = \mathbf{bcd}_1 \quad (4.33a)$$

$$\left(\frac{\mathbf{M}_\rho}{\Delta t} + \lambda\mathbf{K}_\rho\right)\tilde{\mathbf{u}}^{n+1} + \mathbf{G}\tilde{\mathbf{p}}^{n+1} = \left(\frac{\mathbf{M}_\rho}{\Delta t} - (1-\lambda)\mathbf{K}_\rho\right)\tilde{\mathbf{u}}_d^n - \frac{1}{Fr^2}\mathbf{gF}\tilde{\theta}_d^n + \mathbf{bcd}_2 + \mathbf{bcn}_2 \quad (4.33b)$$

$$\left(\frac{\mathbf{M}_\theta}{\Delta t} + \lambda\mathbf{K}_\theta\right)\tilde{\theta}^{n+1} = \left(\frac{\mathbf{M}_\theta}{\Delta t} - (1-\lambda)\mathbf{K}_\theta\right)\tilde{\theta}_d^n + \mathbf{bcd}_3 + \mathbf{bcn}_3 \quad (4.33c)$$

The representation in matrix form of these equations is

$$\begin{bmatrix} \mathbf{M}'_\rho & \mathbf{G} & \mathbf{0} \\ \mathbf{D} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{M}'_\theta \end{bmatrix} \begin{bmatrix} \tilde{\mathbf{u}}^{n+1} \\ \tilde{\mathbf{p}}^{n+1} \\ \tilde{\theta}^{n+1} \end{bmatrix} = \begin{bmatrix} \mathbf{r}_u^n \\ \mathbf{0} \\ \mathbf{r}_\theta^n \end{bmatrix} + \begin{bmatrix} \mathbf{bcd}_2 \\ \mathbf{bcd}_1 \\ \mathbf{bcd}_3 \end{bmatrix} + \begin{bmatrix} \mathbf{bcn}_2 \\ \mathbf{0} \\ \mathbf{bcn}_3 \end{bmatrix} \quad (4.34)$$

where, the matrix  $\mathbf{M}'_\rho$  and  $\mathbf{M}'_\theta$  are given by

$$\mathbf{M}'_\rho = \frac{\mathbf{M}_\rho}{\Delta t} + \lambda\mathbf{K}_\rho \quad (4.35a)$$

$$\mathbf{M}'_\theta = \frac{\mathbf{M}_\theta}{\Delta t} + \lambda\mathbf{K}_\theta \quad (4.35b)$$

and the vector  $\mathbf{r}_u^n$  and  $\mathbf{r}_\theta^n$  are given by

$$\mathbf{r}_u^n = \left(\frac{\mathbf{M}_\rho}{\Delta t} - (1-\lambda)\mathbf{K}_\rho\right)\tilde{\mathbf{u}}_d^n - \frac{1}{Fr^2}\mathbf{gF}\tilde{\theta}_d^n \quad (4.36a)$$

$$\mathbf{r}_\theta^n = \left(\frac{\mathbf{M}_\theta}{\Delta t} - (1-\lambda)\mathbf{K}_\theta\right)\tilde{\theta}_d^n \quad (4.36b)$$

The projection method consists of decomposing the matrix of the equation (4.34) via a block factorization. Applying the LU factorization, the following linear system is obtained

$$\begin{bmatrix} \mathbf{M}'_\rho & \mathbf{0} & \mathbf{0} \\ \mathbf{D} & -\mathbf{D}\mathbf{M}'_\rho^{-1}\mathbf{G} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{M}'_\theta \end{bmatrix} \begin{bmatrix} \mathbf{I} & \mathbf{M}'_\rho^{-1}\mathbf{G} & \mathbf{0} \\ \mathbf{0} & \mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{I} \end{bmatrix} \begin{bmatrix} \tilde{\mathbf{u}}^{n+1} \\ \tilde{\mathbf{p}}^{n+1} \\ \tilde{\theta}^{n+1} \end{bmatrix} = \begin{bmatrix} \mathbf{r}_u^n \\ \mathbf{0} \\ \mathbf{r}_\theta^n \end{bmatrix} + \begin{bmatrix} \mathbf{bc}_2 \\ \mathbf{bc}_1 \\ \mathbf{bc}_3 \end{bmatrix} \quad (4.37)$$

where,  $\mathbf{bc}_i = \mathbf{bcd}_i + \mathbf{bcn}_i$  ( $i = 1, 2, 3$ ).

In the first instance the intermediate solution is obtained by solving the following equation system

$$\begin{bmatrix} \mathbf{M}'_\rho & \mathbf{0} & \mathbf{0} \\ \mathbf{D} & -\mathbf{DM}'_\rho^{-1}\mathbf{G} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{M}'_\theta \end{bmatrix} \begin{bmatrix} \hat{\mathbf{u}}^{n+1} \\ \hat{\mathbf{p}}^{n+1} \\ \hat{\boldsymbol{\theta}}^{n+1} \end{bmatrix} = \begin{bmatrix} \mathbf{r}_u^n \\ \mathbf{0} \\ \mathbf{r}_\theta^n \end{bmatrix} + \begin{bmatrix} \mathbf{bc}_2 \\ \mathbf{bc}_1 \\ \mathbf{bc}_3 \end{bmatrix} \quad (4.38)$$

and then, the final solution is obtained by solving

$$\begin{bmatrix} \mathbf{I} & \mathbf{M}'_\rho^{-1}\mathbf{G} & \mathbf{0} \\ \mathbf{0} & \mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{I} \end{bmatrix} \begin{bmatrix} \tilde{\mathbf{u}}^{n+1} \\ \tilde{\mathbf{p}}^{n+1} \\ \tilde{\boldsymbol{\theta}}^{n+1} \end{bmatrix} = \begin{bmatrix} \hat{\mathbf{u}}^{n+1} \\ \hat{\mathbf{p}}^{n+1} \\ \hat{\boldsymbol{\theta}}^{n+1} \end{bmatrix} \quad (4.39)$$

Using the two equations (equation 4.38 and 4.39) the solution can be obtained by the following steps

$$\mathbf{M}'_\rho \hat{\mathbf{u}}^{n+1} = \mathbf{r}_u^n + \mathbf{bc}_2 \quad (4.40a)$$

$$\mathbf{DM}'_\rho^{-1}\mathbf{G}\tilde{\mathbf{p}}^{n+1} = \mathbf{D}\hat{\mathbf{u}}^{n+1} - \mathbf{bc}_1 \quad (4.40b)$$

$$\mathbf{M}'_\theta \tilde{\boldsymbol{\theta}}^{n+1} = \mathbf{r}_\theta^n + \mathbf{bc}_3 \quad (4.40c)$$

$$\tilde{\mathbf{u}}^{n+1} = \hat{\mathbf{u}}^{n+1} - \mathbf{M}'_\rho^{-1}\mathbf{G}\tilde{\mathbf{p}}^{n+1} \quad (4.40d)$$

This method relies on the Helmholtz-Hodge decomposition, which says that any vector can be decomposed into a component of a zero divergence and another with zero curl.

Solving the equations (4.40) is known as the Uzawa method. However, to solve the equation (4.40b) exactly is a very expensive step. Therefore, an approximation is performed in order to increase the computational efficiency, yielding the following approximate factorization

$$\mathbf{M}'_\rho \hat{\mathbf{u}}^{n+1} = \mathbf{r}_u^n + \mathbf{bc}_2 \quad (4.41a)$$

$$\mathbf{D}\widetilde{\mathbf{M}}_\rho'^{-1}\mathbf{G}\tilde{\mathbf{p}}^{n+1} = \mathbf{D}\hat{\mathbf{u}}^{n+1} - \mathbf{bc}_1 \quad (4.41b)$$

$$\mathbf{M}'_\theta \tilde{\boldsymbol{\theta}}^{n+1} = \mathbf{r}_\theta^n + \mathbf{bc}_3 \quad (4.41c)$$

$$\tilde{\mathbf{u}}^{n+1} = \hat{\mathbf{u}}^{n+1} - \widetilde{\mathbf{M}}_{\rho}'^{-1} \mathbf{G} \tilde{\mathbf{p}}^{n+1} \quad (4.41d)$$

where  $\widetilde{\mathbf{M}}_{\rho}'$  is a diagonal (lumped) approximation of  $\mathbf{M}_{\rho}'$

## CHAPTER 5

---

### SYSTEM DESIGN

---

Object Oriented Programming (OOP) may provide a higher abstraction level that simplifies the modeling of the otherwise complex interactions between model components. The Unified Modeling Language (UML) is a standardized modeling language to describe object oriented projects. The UML is a graphical language to visualize and document the artifacts of a system design. UML has several types of graphical diagrams including the class diagram that is one of the most popular. The class diagram describes the structure of a system by showing its classes, their attributes, and their relationships. The figure 5.1 is a class diagram showing the relation between the *GUI* class and the classes used to manipulated different terrain models, as well as, mesh generators. Just some basic attributes and operations are displayed.

The *GUI* class is responsible to manage the graphical interface which allows user to interact with the different components of the simulation software. The *GUI* class has areas to visualize different objects. The *areaTerrain* object contains information of the terrain objects derived from the *clShapeTerrain*, *clCloudTerrain* and *clRasterTerrain* classes. The *clShapeTerrain* class is responsible to read the shapefile and to store the contour lines data. The *clCloudTerrain* class extracts the points of the contour lines data of the object of the *clShapeTerrain* class received. *clCloudTerrain*, also can read point data files. The *clRasterTerrain* class reads two types of file: DTM data files and GeoTIFF files. This class stores the pixel information into a

matrix in *RGBA* color format.

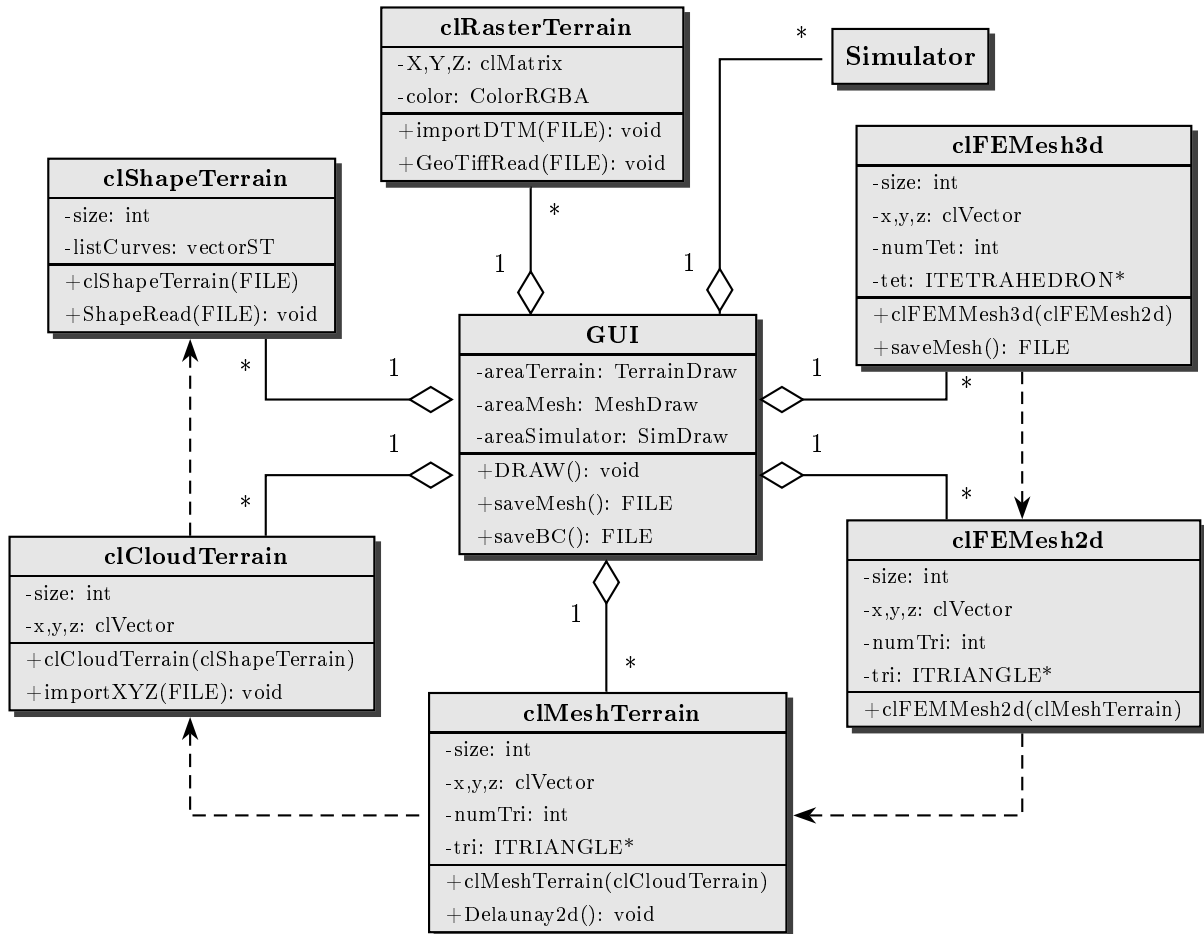
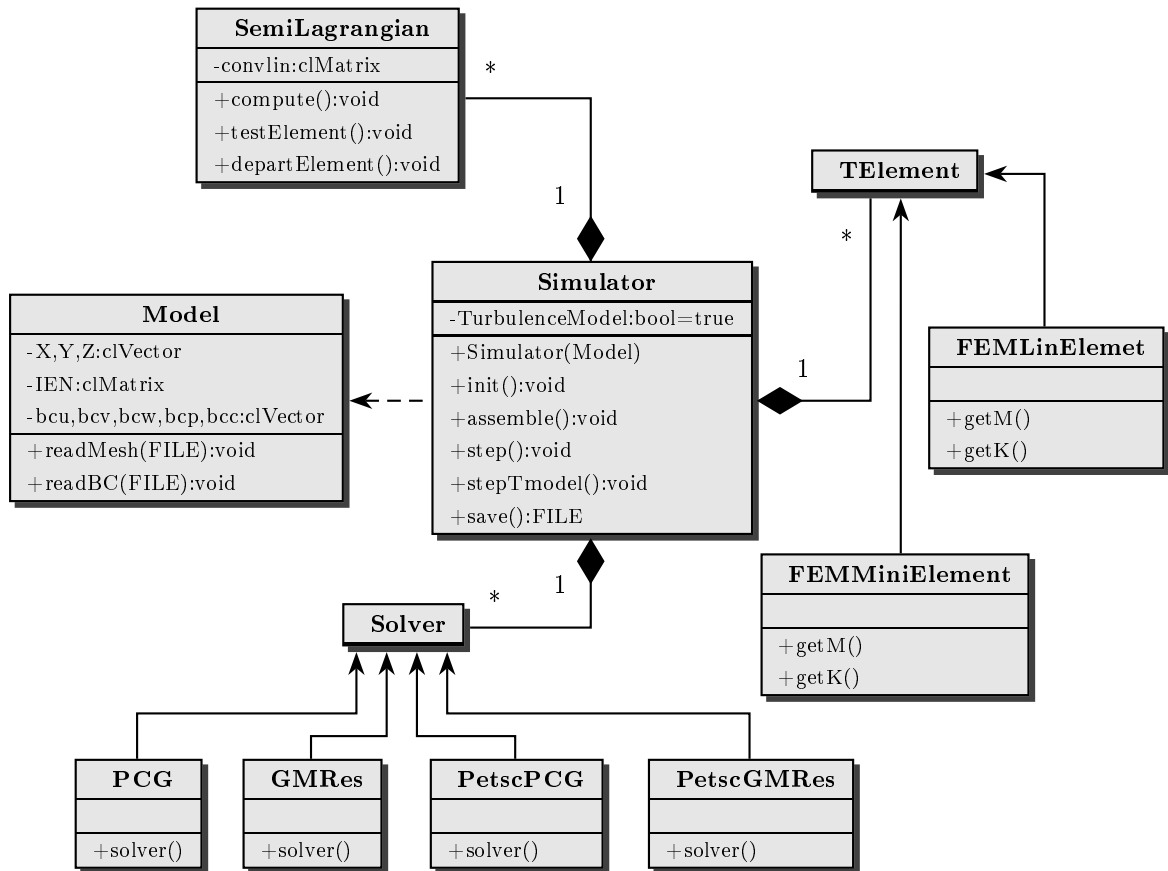


Figure 5.1: UML class diagram of the *GUI*.

The *areaMesh* object contains information of the mesh object derived from the *clMeshTerrain*, *clFEMesh2d* and *clFEMesh3d* classes. The *clMeshTerrain* class receives a *clCloudTerrain* object and generates the two dimensional mesh using the Delaunay triangulation algorithm. The *clFEMesh2d* class bounds the domain of a object of the *clMeshTerrain* class to obtain a two dimensional finite element mesh. The *clFEMesh3d* class is responsible to generate the three dimensional finite element mesh from a object of the *clFEMesh2d* class.

At last, the *areaSim* object stores the information of the objects of the *Simulator* class. The *GUI* class provides methods that allow the visualization of the objects described and to set the different conditions for the simulation. The details of the *Simulator* class are shown in the figure 5.2.

The *Simulator* class is responsible to interconnect the various classes for the simulation.

Figure 5.2: UML class diagram of the *Simulator*.

The constructor of the *Simulator* class receives a object of the *Model* class. This class reads the files (mesh, boundary conditions and simulation option) which contain the information necessary for simulation and stores it. The value of the *TurbulenceModel* determines whether the turbulence model is used or not. The matrices are assembled by the member *assemble* of the *Simulator* class. This member uses the *TElement* class, that depending of the type of element, linear tetrahedron or mini tetrahedron, is called the heir *FEMLinElement* or *FEMMiniElement* classes. The *stepTmodel* member of the *Simulator* class uses the object of the *SemiLagrangian* class, that mounts the *convLin* matrix to compute the effect of the substantial derivative operator. The objects of the *Solver* class are used to solve the linear equations system obtained. If the matrix of the linear system is symmetric positive defined (SPD), the object of the *PCG* class is used. But, if the matrix is not positive defined, the object of the *GMRes* class is used. The *PetscPCG* and *PetscGMRes* classes use the Portable Extensible Toolkit for Scientific Computation (PETSc) (BALAY et al., 2009; BALAY et al., 2008; BALAY et al., 1997), that is a data structures and routines library based on the Message Passing Interface Implementation (mpich),



---

Basic Linear Algebra Subprograms (BLAS) and Linear Algebra Package (LAPACK). At last, the solutions are saved in ASCII format by the *save* member of the *Simulator* class.

## CHAPTER 6

---

# NUMERICAL SIMULATIONS

---

The code validation was performed comparing the results of simulations with results from the literature and analytical solutions of flows with uniform density.

In this chapter, the numerical simulations of gravity currents are shown. Several numerical simulations have been performed to compare with the results of 2D gravity currents flows available in the literature. Then, the numerical simulations of the scale model of a confluence and the flow in a branch of a reservoir are shown.

Gravity current is a fluid flow driven by density difference. The structure of the gravity current is affected by the stratification. Since the gravity currents are formed in several natural phenomena and also man-made, several experiments and simulations have been done to better understand it.

The simulations are performed in server with the following configurations:

- 2 Quad-Core Intel Xeon processor E5320 1.86 *GHz* 8 *MB* 1066 *MHz*
- RAM: 16 *GB*
- HD: 400 *GB*
- 2 Port Giga Ethernet
- OS: Ubuntu Server 8.04

## 6.1 2D Gravity Current

The results of the 2D gravity current simulation are shown. The configuration of the numerical simulation is the same as the laboratory experiment performed by Eghbalzadeh et al. (2008). In the experiments of Eghbalzadeh et al. (2008), a rectangular channel was filled with fresh water with a depth of  $0.1\text{ m}$ . A gate is located at a distance of  $0.3\text{ m}$  from the left end dividing the channel into two regions (Figure 6.1). Salt (NaCl) was added to the left until the uniform density  $1011.4\text{ kg/m}^{-3}$  was obtained. Finally, the gate was removed suddenly.



Figure 6.1: Initial configuration of the 2D gravity current simulation.

The spatial computational domain used is a parallelepiped with the following dimension: height  $0.1\text{ m}$ , length  $1.2\text{ m}$  and  $0.01\text{ m}$  thickness. Vertical faces and the bottom horizontal face are considered as solid surface and the top horizontal face as free surface. The salinity and the density employed in the numerical simulation are based on the *One Atmosphere International Equation of State of Seawater* (UNESCO, 1981). The density of fresh water at  $20^\circ\text{C}$  is  $\rho_r = 998.2063\text{ kg/m}^{-3}$  and the salinity to obtain the density of the salt water released in the experiment is  $17.41\text{ g/l}$ . The reduced gravity (equation 3.66) is  $g' = 0.1297\text{ m/s}^2$  and the contraction coefficient due to the salinity is  $\beta = 7.5919 \times 10^{-4}\text{ m}^3/\text{kg}$ . The Schmidt number using the molecular values (molecular viscosity and diffusivity) is  $Sc = 640$ . The turbulent Schmidt number were  $Sc_t = 2$ .

The mesh has 10 406 vertices points and 30 240 tetrahedral elements and the time step adopted is  $t_s = 0.01074\text{ s}$ .

The characteristic length adopted is the half the height of the channel, that is  $L_c = 0.05\text{ m}$ . The velocity is scaled by the wave velocity at the interface defined as  $U_c = \sqrt{g'L_c} = 8.0518 \times 10^{-2}\text{ m/s}$ . Then, the characteristic time is  $t_c = \frac{L_c}{U_c} = 0.621\text{ s}$ . The Reynolds number, using molecular viscosity ( $\nu = 1.01 \times 10^{-6}\text{ m}^2/\text{s}$ ), is  $Re = 4026$ .

The linear equation system is solved using data structures and solvers provided by PETSc. The matrices  $M'_\rho$  and  $M'_\theta$  of the equations (4.41a) and (4.41c), respectively, are symmetric positive defined. Then, these equations are solved using Preconditioned Conjugate Gra-

dient (PCG) iterative method. The matrix  $\widetilde{\mathbf{M}}_\rho^{-1}$  of the equation (4.41b) is diagonal, but the matrices  $\mathbf{D}$  and  $\mathbf{G}$  are not transpose to each other. Then, the matrix  $\mathbf{D}\widetilde{\mathbf{M}}_\rho^{-1}\mathbf{G}$  is not symmetric. For this reason, the equation (4.41b) is solved using Generalized Minimal Residual (GMRes) method.

Comparison between the results obtained by Eghbalzadeh et al. (2008) and the simulation performed are shown in the figure 6.2. The upper picture is the visualization of the laboratory experiment, the intermediate picture following corresponds to the numerical simulation carried out by Eghbalzadeh et al. (2008) for coarse grid resolution, and the lower picture is the visualization of the numerical simulation performed.

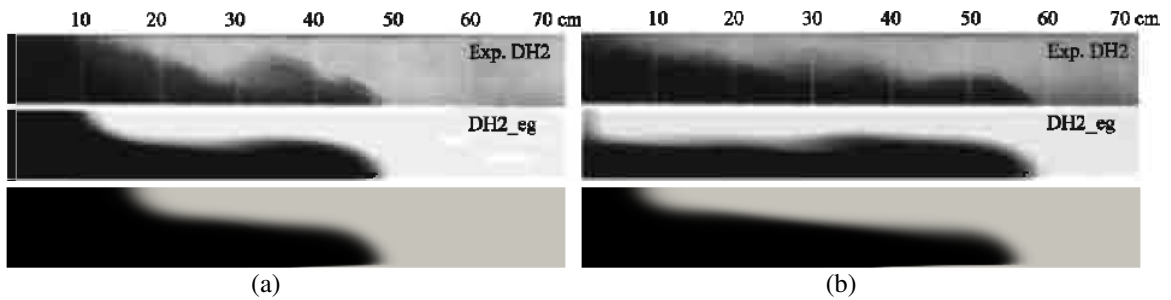


Figure 6.2: Comparison between the experimental and numerical results of Eghbalzadeh et al. (2008) and the simulation performed: (a)  $t = 4.4$  s. (b)  $t = 6.8$  s

Figure 6.3 shows the velocity of the gravity current nose computed from the numerical simulation. There is good correlation at the beginning of the simulation. There is no significant

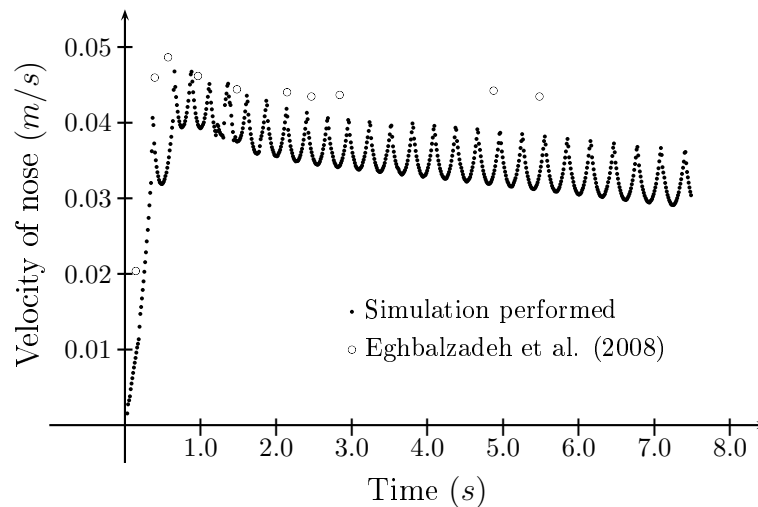


Figure 6.3: Velocity of the computed gravity current nose.

difference comparing the initial acceleration and the peak velocity reached by the nose for both simulations. After the stage of acceleration, velocity decays in the simulation possibly due to numerical diffusion. This numerical diffusion is caused by coarse mesh resolution and the low order (only first order) discretization employed in the semi-Lagrangian method.

## 6.2 Model of confluence

A confluence of rivers is the point where two or more rivers join, bodies of water. The bodies of water that meet may have different properties such as temperature, concentration of salts and other compounds, sediments, pollutants, among others. Stable stratification can maintain different bodies of water as separate, with little mixing. In contrast, the turbulence can break up the stratification and generate strong mixing of the bodies of water. Understanding the physics of the confluence is very important for the prediction of impacts that it may produce.

The gravity current flow on a scale model of a confluence is studied by means of laboratory experiment and numerical simulation, in order to validate the code in the case of complex geometry.

### 6.2.1 Laboratory Experiment

The Figure 6.4 show the scale model of confluence used. Salt water was released from behind a lock gate placed on the tributary (point A of the Figure 6.4). The lines showed in Figure 6.4(a) are contour lines of water depth.

The length of the main stem is  $1.195\text{ m}$  and the width is  $0.5\text{ m}$ ; the length of the tributary is  $0.5\text{ m}$  and the width is  $0.1\text{ m}$ . The lock gate is placed at  $0.1\text{ m}$  from the confluence. The height of the water level is stabilized at  $0.06\text{ m}$ , measured on the gate. The temperature of the water is  $25\text{ }^\circ\text{C}$  and the salinity of  $4\text{ g/l}$  for the salt water on the tributary is used. The reduced gravity is  $g' = 0.296\text{ m/s}^2$  and the contraction coefficient due to the salinity is  $\beta = 7.5421 \times 10^{-4}\text{ m}^3/\text{kg}$ .

### 6.2.2 Numerical Simulation

The mesh forming the computational domain is generated from a cloud of points that lay in 21 contour lines separated on average  $4.2\text{ mm}$  in the vertical direction. At first, the two-

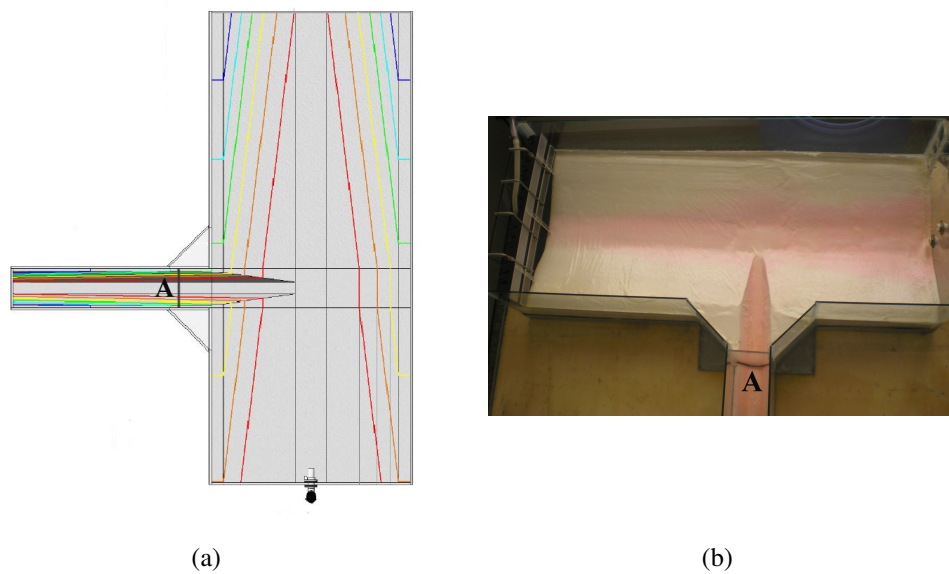


Figure 6.4: Scale model of confluence: (a) Sketch. (b) Picture.

dimensional surface mesh is generated (figure 6.5(a)). In the main stem the points are distributed uniformly, but in the tributary a nonuniform distribution was introduced applying the stretching defined by the equation 2.8, for  $\phi = 90^\circ$  and  $\eta = 3$ . Then, the tetrahedral mesh is generated using the algorithms described in section 2.2. The mesh generated has 49 762 vertices, 311 987 node points and 262 225 tetrahedral elements. Notice that the node points are associated with vertices plus the centroid (bubble node) of each element. The figure 6.5(b) shows a close-up of the 3D mesh obtained.

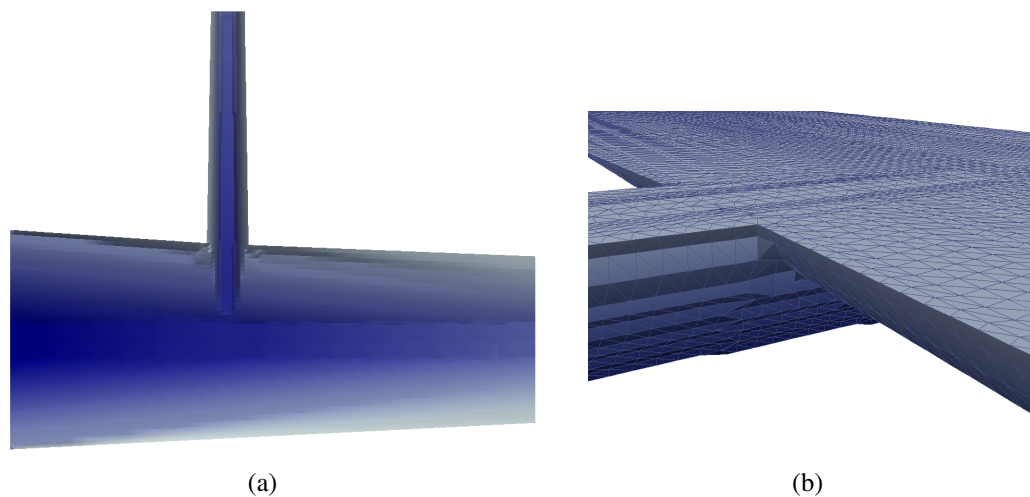


Figure 6.5: Mesh of the confluence model: (a) Surface mesh. (b) 3D mesh with edges of the tetrahedra.

### 6.2.3 Results

Laboratory and numerical simulations were performed for the same choice of parameters and initial conditions. The laboratory experiment was recorded on digital video and the captured images were compared with visualizations obtained from the data of the numerical simulation at similar times.

The laboratory experiment and the numerical simulation both show three main periods:

- During a first stage, a gravity wave propagates along the tributary similarly to the 2D gravity current in the section 6.1. Figure 6.6 shows the development of the 2D wave along the tributary.



Figure 6.6: Numerical simulation of gravity current wave propagating along the tributary.

- On the second stage, the gravity current reaches the main stem and propagates similarly to an axisymmetric gravity current (figure 6.7).

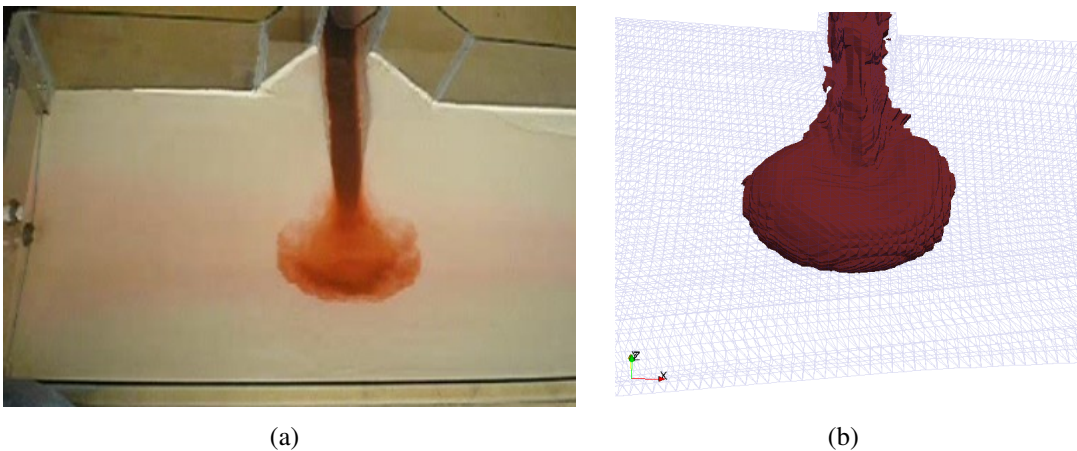


Figure 6.7: Propagation of the wave similar to an axisymmetric gravity current: (a) Laboratory experiment. (b) Numerical simulation.

- Finally, on a third stage, the current interacts with the bottom of the main stem and starts to propagate upstream and downstream of the main stem (figure 6.8 and 6.9).

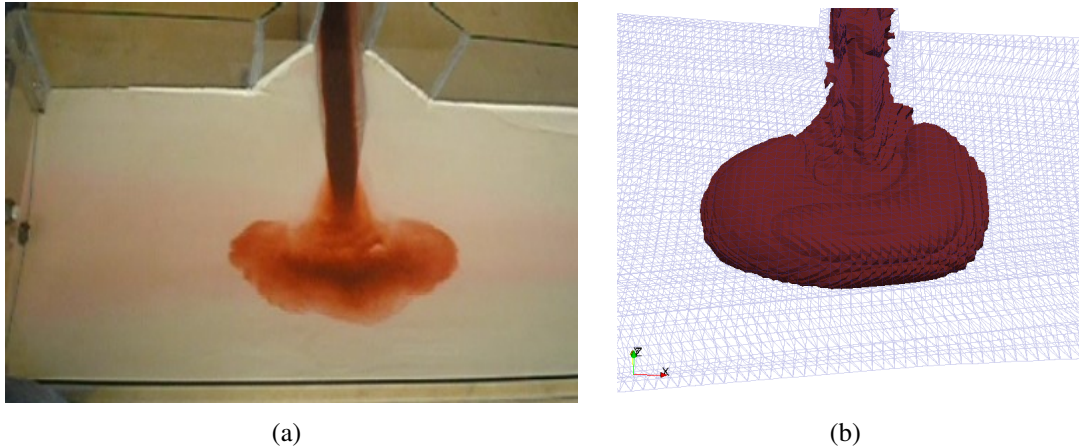


Figure 6.8: Propagation of the wave in the main stem: (a) Laboratory experiment. (b) Numerical simulation.

Figures 6.7 to 6.9 show good correlation between experimental and numerical results.

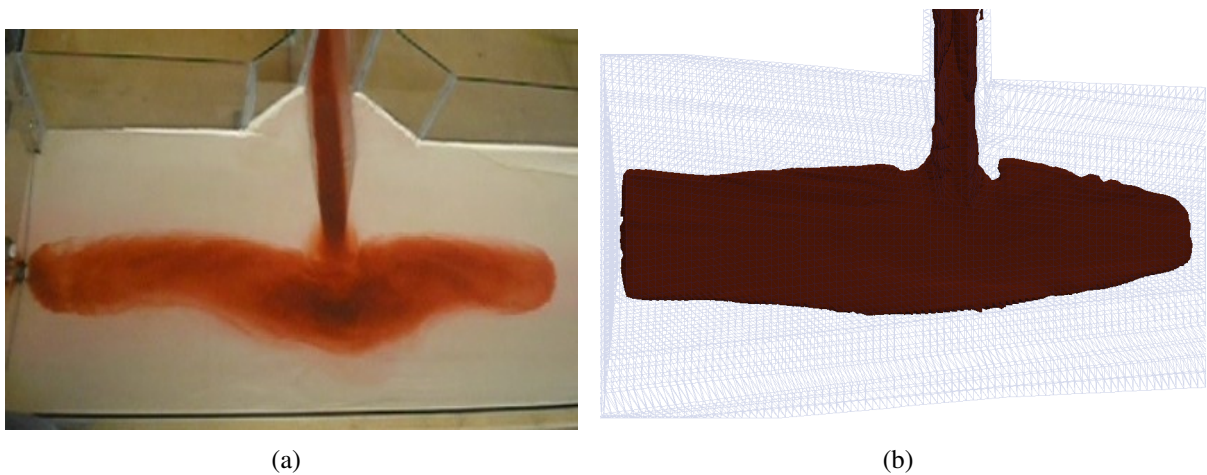


Figure 6.9: Propagation of the wave in the main stem: (a) Laboratory experiment. (b) Numerical simulation.

### 6.3 Reservoir Simulation

To perform a numerical simulation of a branch of a natural reservoir the shapefile containing the geographic data of a reservoir is loaded and a branch is selected. The contour lines correspond to a region in Rio de Janeiro state (Brazil) close to  $42.577^\circ$  S –  $22.415^\circ$  W. The



contour lines separations are  $20.0\text{ m}$  in vertical direction. Figure 6.10 shows the contour lines with the dimensions of the reservoir. The cloud of points is extracted from the contour lines. Figure 6.11 shows the contour lines and the cloud of points lying in these contour lines.

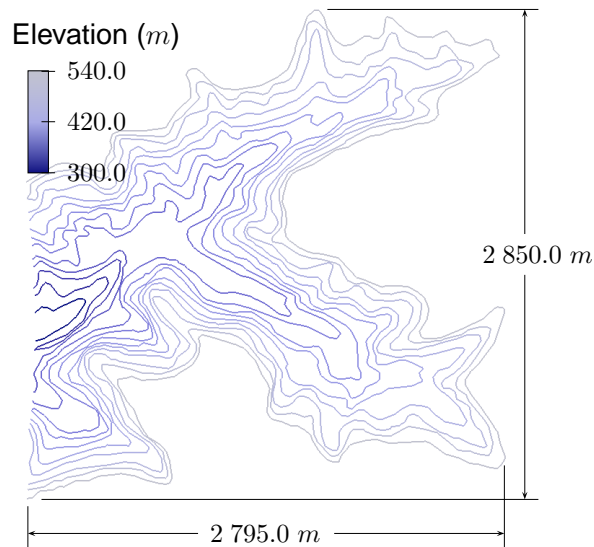


Figure 6.10: Contour lines and the dimensions of the reservoir *GUI*.

From the cloud of points, using the Delaunay triangulation algorithm, the surface mesh is generated. Figure 6.12 shows the 2D image of the triangular mesh shown in the graphic user interface (*GUI*) and the Figure 6.13(a) shows the triangular surface mesh. By adding points into the reservoir domain and using tetrahedra construction from the triangular prism, the tetrahedra mesh is generated (Figure 6.13(b)).

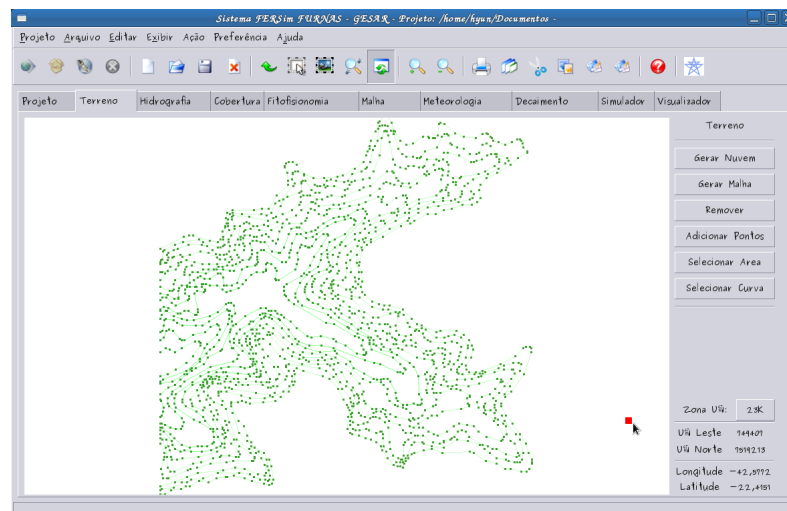
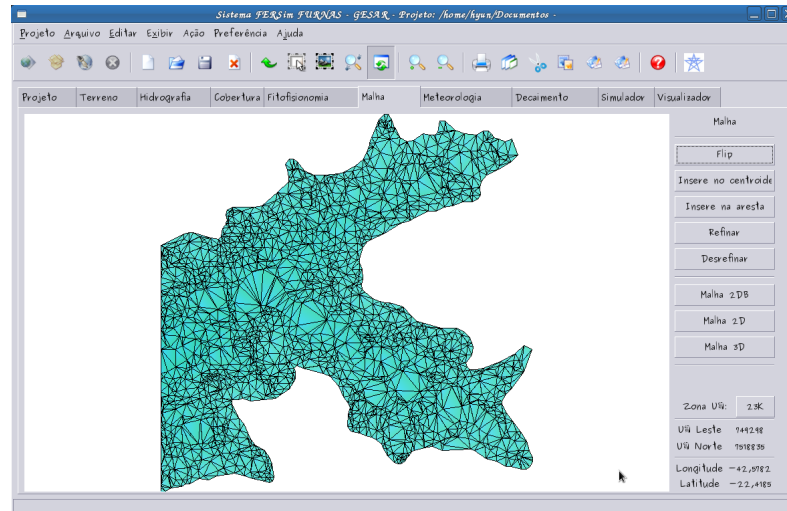


Figure 6.11: Contour lines and cloud of point shown in the interface *GUI*.

Figure 6.12: 2D triangular mesh shown in the *GUI*.

The Figures 6.14 to 6.19 show different stages of the simulation. Figure 6.14 shows the initial conditions of the simulation. The simulation has two inflows that meet in a confluence. The inflow velocity of the left side (inflow A) of the Figure 6.14 is  $0.6708 \text{ m/s}$  with null scalar concentration and the right side inflow velocity (inflow B) is  $0.6403 \text{ m/s}$  with 3% of concentration. The gray surface is an isosurface of scalar concentration corresponding to a value of 0.1%. The size and the color of the arrows are related to the magnitude of the velocity and the direction of the arrows is related to the direction of velocity. The color of the arrow is limited by red for maximum velocity and blue for minimum velocity, respectively. The initial scalar concentration is stratified and advances due to the velocity caused by the inflow and also due to

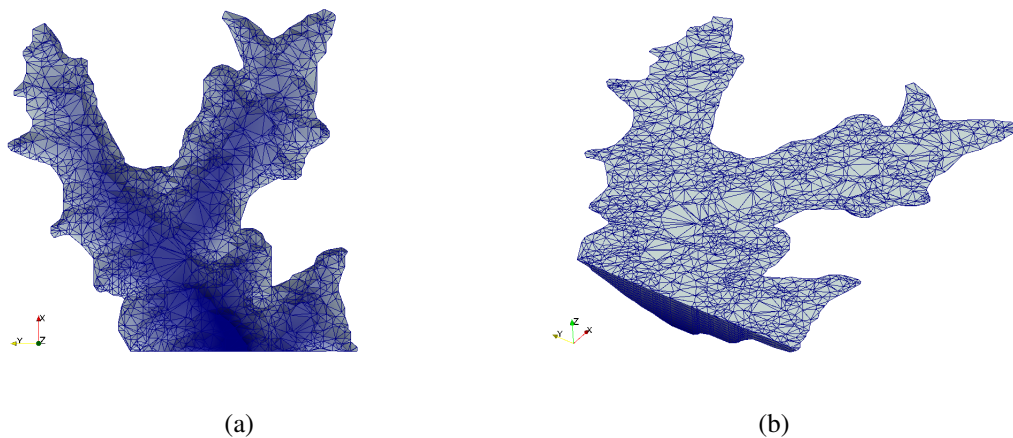


Figure 6.13: Mesh of the reservoir: (a) Surface mesh (top view). (b) 3D mesh.

a gravity current that is emphasized by the downstream slope given by the geographic condition.

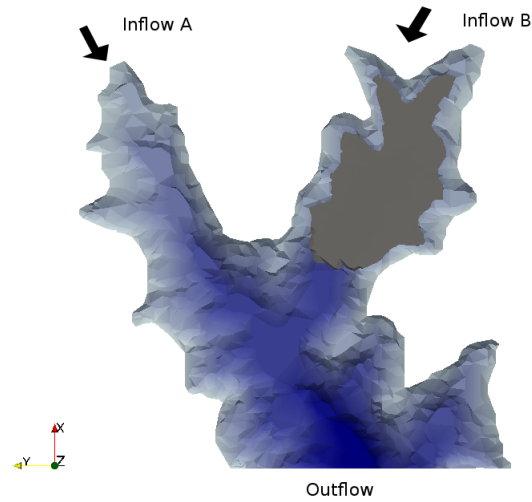


Figure 6.14: Initial condition of the simulation.

Figure 6.15 shows the simulation at 14.87 s. The scalar advances mainly along the bottom of reservoir and there is very little diffusion.

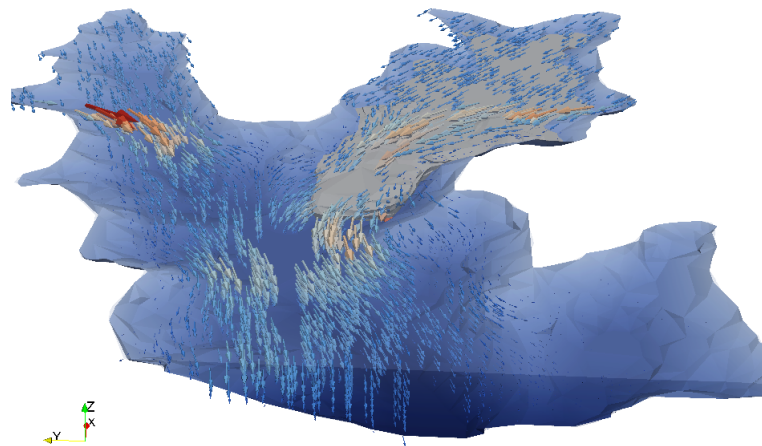


Figure 6.15: Simulation at 14.87 s.

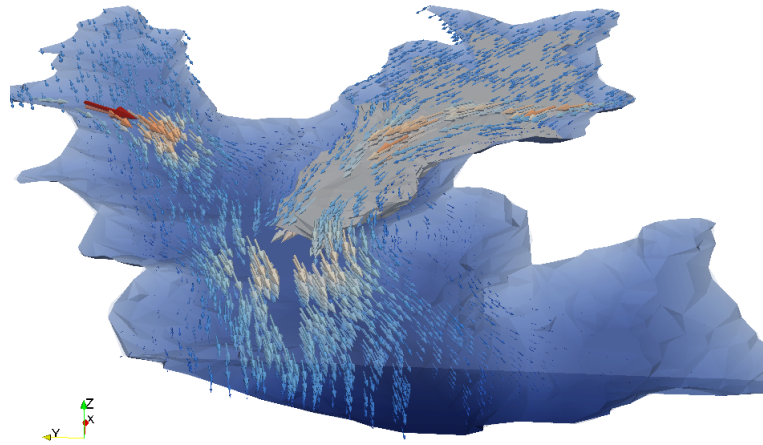


Figure 6.16: Simulation at 29.74  $s$ .

The figures 6.16 (simulation at time step 29.74  $s$ ) and 6.17 (simulation at time step 59.48  $s$ ) show the fast advance of the scalar front. Notice that the passage of the fluid from inflow A is disturbed due to the inertia of the gravity current coming from the inflow B, which is caused by density gradients.

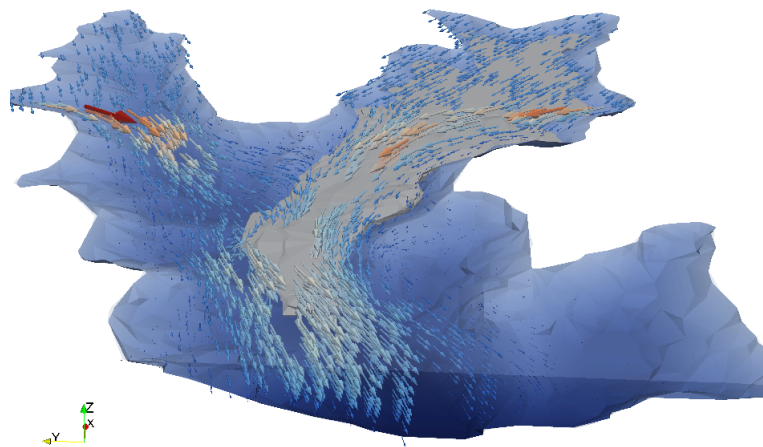
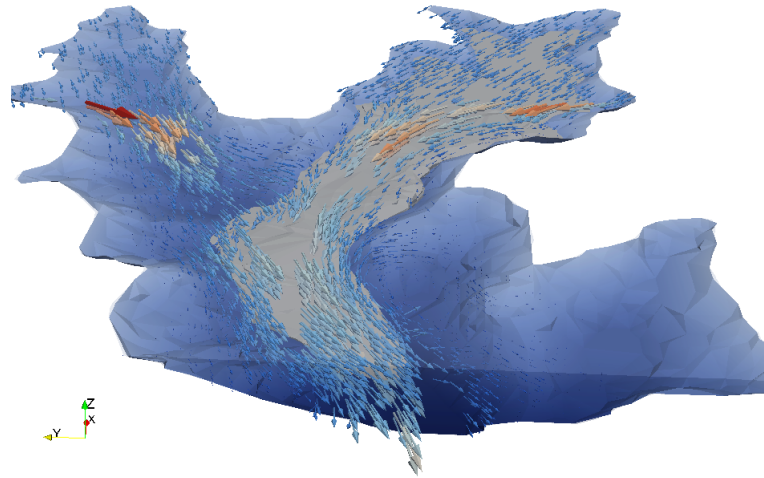
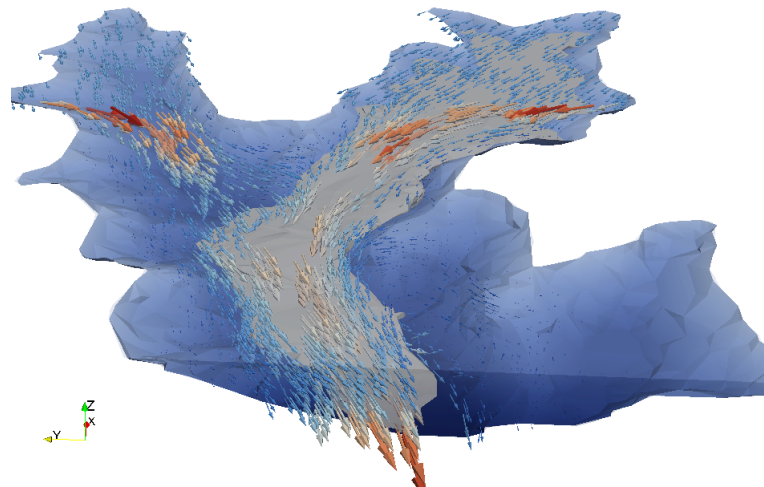


Figure 6.17: Simulation at 59.48  $s$ .

Figure 6.18: Simulation at 89.22  $s$ .

The figure 6.18 (simulation at time step 89.22  $s$ ) shows the isosurface of scalar concentration almost reaching the zone of outflow. It is possible to observe that the velocity field has focused even more on the area where the scalar advances. The figure 6.19 (simulation at time step 178.44  $s$ ) shows that the scalar isosurface reached the outflow zone. After this stage, there are no significant variations in the flow.

Figure 6.19: Simulation at 178.44  $s$ .

The flow of fluid within the isosurface of scalar concentration constitutes a gravity current. The heavier fluid moves along the deepest regions of the reservoir, with a much higher velocity than the surface layers. Thus producing a short circuit along the bottom of the reser-

voir. This phenomenon is important because it substantially decreases the mean residence time of the water leaving the reservoir. The simulated behavior is consistent with the behavior observed in reservoirs under similar conditions.

The simulation of the flow in a reservoir shows a flow pattern that is qualitatively consistent with field observation. The results confirm that the code is capable of dealing with complex geometry and physics typical of real applications in reservoir modeling.

# CHAPTER 7

---

## SUMMARY

---

This work is part of a project, which proposes the construction of a methodology to study and forecast the environmental impacts that a hydroelectric power plant reservoir may produce. This project is developed by a group called GESAR and joins different areas of science such as mechanical engineering for fluid flow analysis, mathematics for resolution of the discretized equations of fluid flow, computational science for computational resource to simulate the fluid flow, geography for geographic information system (GIS) data analysis, biology and chemistry for biomass decomposition and water quality analysis.

In this context, the objectives of this work are to provide numerical simulation tools for 3D environmental flows and to describe a methodology for numerical simulation of the reservoir. In the numerical simulation of hydrodynamic applications in real geophysical system, the areas cited above are involved. This work describes not only the equations and the numerical methods of the fluid flow but also several issues, that are related to the numerical simulation of the reservoir.

A tetrahedral mesh generator of the reservoir based on the terrain topology and an algebraic turbulence model based on the Richardson number are the main tools developed.

The chapter 5 gives information on the code structure based on the object oriented programming. The entire code was developed by the group of numerical simulation of GESAR,

that the author belongs to. The classes *clShapeTerrain*, *clCloudTerrain*, *clRasterTerrain* and *clMeshTerrain* were previously described by Mangiavacchi et al. (2005), Dongala et al. (2006). Part of the classes *clMeshTerrain*, *clFEMesh2d* and *clFEMesh3d* referring to the mesh generation were developed by the author for this work. The class *Simulator*, originally described by Anjos et al. (2007), is the hydrodynamic and scalar transport simulator. In this work, it was introduced a turbulence model for stratified flow. It is important to emphasize that the code presented is only one part of all the code developed by the group, because there are other modules such 2DH, 2DB simulation, decay and meteorology which are not represented in this work.

For numerical simulation of real geophysical systems it is necessary to represent adequately the terrain geophysical topology. The points used by the mesh generator are extracted from the terrain topological data. The main difficulty in tetrahedral mesh generation of a reservoir is non-uniform distribution of the points related to the huge ratio between the horizontal and vertical scales of the reservoir. In this type of points distribution, conventional tetrahedron mesh generation algorithm may become unstable. For this reason, a unstructured tetrahedral mesh generator was developed and the methodology used was described in detail in this work. Triangular surface mesh generation using the Delaunay triangulation and the construction of the tetrahedra from the triangular surface information are the main steps to the mesh generator. To generate the triangular surface mesh a randomized incremental algorithm with optimized search was used. Furthermore, the triangular surface mesh generator is able to obtain meshes with edges having a preferential direction. This algorithm is used to generate the mesh of the tributary of the confluence model (section 6.2). From the triangular surface mesh, by adding points into the computational domain and prism partition process, the tetrahedral mesh was generated. The points are added respecting the layers of the contour lines and sticks formed by the vertical lines passing through the points at the surface. With this methodology, it was possible to construct prisms and split them into conforming tetrahedra.

Direct Numerical Simulation of the Navier-Stokes equations is very important for flow analysis. But this technique requires large computational resources, which in practical engineering situations are not feasible. The hydrodynamic simulation with a turbulence model provides a useful tool that is computationally viable for engineering purposes. Eddy-Viscosity/Diffusivity models for stratified turbulent flows were incorporated in the Reynolds-averaged Navier-Stokes (RANS) and mean scalar transport equations. These models are algebraic models based on the



Richardson number and provide realistic results with the fitting of a small amount of parameters.

Several numerical simulations have been performed to compare with the results available in the literature. Comparisons with the results obtained by Eghbalzadeh et al. (2008) for the 2D gravity current show that there is good correlation at the beginning of the simulation, related to the initial acceleration and the peak velocity reached by the gravity current nose. After the stage of acceleration, the velocity decays in the performed simulation possibly due to numerical diffusion caused by coarse mesh resolution and the low order (only first order) discretization employed in the semi-Lagrangian method.

The simulation of the model of confluence is performed in order to compare the structure of the 3D gravity current with the laboratory experiment. The results obtained were satisfactory, considering that a relatively coarse mesh, limited by the computational resources, has been used.

Finally, the simulation of a branch of a reservoir was adequate to verify the mesh generator from the real terrain data and also the simulation of the flow in a potential reservoir application.

## 7.1 Future Research Area

In this work, 3D reservoir simulation with algebraic turbulent model was tested and validated.

A number of improvements deserve further attention in order to achieve better performance of the simulation tool.

In terms of the mesh generator, the uses of an adequate data structure to save the information of the mesh may provide savings in the computational cost.

In terms of the physics, considering that the meteorological data is a factor that is closely related to the stratification and mixing process, it is necessity to link the meteorological information with the 3D fluid flow simulation.

In terms of the computational resource, the problem of the numerical diffusion may be reduced by developing high order semi-Lagrangian methods. High performance scientific computing techniques must be developed and employed to allow the use of finer meshes and provide more accuracy in the simulations.

---

## REFERENCES

---

ANJOS, G. R. d. et al. Numerical modeling of the hydrodynamic field coupled to the transport of chemical species through the finiteelement. In: *6th International Congress on Industrial and Applied Mathematics (ICIAM 2007)*. Zuriq: 6th International Congress on Industrial and Applied Mathematics, 2007.

BALAY, S. et al. *PETSc Users Manual*. [S.l.], 2008.

BALAY, S. et al. *PETSc Web page*. 2009. [Http://www.mcs.anl.gov/petsc](http://www.mcs.anl.gov/petsc).

BALAY, S. et al. Efficient management of parallelism in object oriented numerical software libraries. In: ARGE, E.; BRUASET, A. M.; LANGTANGEN, H. P. (Ed.). *Modern Software Tools in Scientific Computing*. [S.l.]: Birkhäuser Press, 1997. p. 163–202.

BATCHELOR, G. *An introduction to fluid dynamics*. UK: Cambridge University Press, 2000.

BATES, P. D.; LANE, S. N.; FERGUSON, R. I. *Computational Fluid Dynamics: Applications in Environmental Hydraulics*. England: John Wiley & Sons Ltd, 2005. 1–15 p.

BERG, M. de et al. *Computational Geometry: Algorithms and applications*. 2nd rev.. ed. Berlin: Springer, 2000.

- DEMARTY, M.; BASTIEN, J.; TREMBLAY, A. Carbon dioxide and methane annual emissions from two boreal reservoirs and nearby lakes in quebec, canada. *Biogeosciences Discussions*, v. 6, n. 2, p. 2939–63, 2009.
- DONGALA, A. M. et al. Finite element mesh generation for numerical simulation of hydroelectric power plant reservoir filling. In: *11th Congress of Thermal Sciences and Engineering (ENCIT 2006)*. Curitiba: 11th Congress of Thermal Sciences and Engineering, 2006.
- EGHBALZADEH, A. et al. Urans simulation of 2d continuous and discontinuous gravity currents. *Journal of Applied Sciences*, v. 8, n. 16, p. 2801–13, 2008.
- FERZIGER, J. H.; PERIĆ, M. *Computational methods for fluid dynamics*. 3rd rev.. ed. Berlin, Heidelberg, New York: Springer-Verlag, 2002.
- GUÉRIN, F. et al. Methane and carbon dioxide emissions from tropical reservoirs: Significance of downstream rivers. *Geophys. Res. Lett.*, v. 33, 2006.
- HARMON, J. E.; ANDERSON, S. J. *The design and implementation of geographic information systems*. New Jersey: John Wiley & Sons, Inc., 2003.
- HEINRICH, J. C.; PEPPER, D. W. *Intermediate finite element method: Fluid flow and heat transfer applications*. NY: Taylor & Francis, 1999.
- HUGHES, T. *The finite element method: Linear static and dynamic finite element analysis*. New York: Dover Publications, 2000.
- KUNDU, P. K.; COHEN, I. M. *Fluid Mechanics*. 2nd. ed. London and San Diego, CA: Academic Press, 2002.
- LANDAU, L. D.; LIFSHITZ, E. M. *Fluid Mechanics*. 2nd. ed. Great Britain: Pergamon Press, 1987.
- LEE, R. C.; TEPFENHART, W. M. *UML and C++: A practical guide to object-oriented development*. Upper Saddle River, NJ, USA: Prentice Hall PTR, 2000.

LEWIS, R.; NITHIARASU, P.; SEETHARAMU, K. *Fundamentals of the finite element method for heat and fluid flow*. UK: John Wiley & Sons, Ltd, 2004.

MANGIAVACCHI, N. et al. Object oriented methodology for the numerical modeling of terrain data for hydrodynamic applications. In: *VIII Encontro de Modelagem Computacional (VIII EMC)*. RJ: VIII Encontro de Modelagem Computacional, 2005.

MAVRIPLIS, D. J. Unstructured grid techniques. *Annual Review of Fluid Mechanics*, v. 29, p. 473–514, 1997.

OERTEL, H. (Ed.). *Prandtl's essentials of fluid mechanics*. 2nd. ed. New York: Springer-Verlag New York, Inc., 2004. (Applied Mathematical Sciences).

O'ROURKE, J. *Computational Geometry in C*. Cambridge: Cambridge University Press, 1998.

PIRONNEAU, O. *Finite element methods for fluids*. Paris: John Wiley & Sons and Masson, 1989.

POPE, S. B. *Turbulent Flows*. Cambridge and New York: Cambridge University Press, 2000.

ROBERT, A.; YEE, T.; RITCHIE, H. A semi-lagrangian and semi-implicit numerical integration scheme for multilevel atmospheric models. *Monthly Weather Review*, v. 113, p. 388–94, 1984.

SCHLICHTING, H.; GERSTEN, K. *Boundary layer theory*. 8th rev. ed. Berlin: Springer-Verlag, 2000.

SHEWCHUK, J. R. What is a good linear element? interpolation, conditioning, and quality measures. In: *In 11th International Meshing Roundtable ed Sandia National Laboratories*. New York: [s.n.], 2002. p. 115–26.

SOTIROPOULOS, F. Introduction to statistical turbulence modeling for hydraulic engineering flow. In: BATES, P. D.; N.LANE, S.; FERGUSON, R. I. (Ed.). *Computational Fluid Dynamics: Applications in Environmental Hydraulics*. England: John Wiley & Sons Ltd, 2005. cap. 5, p. 91–120.

TENNEKES, H.; LUMLEY, J. *A first course in turbulence*. Cambridge, MA: MIT Press, 1972.

TRITTON, D. *Physical fluid dynamics*. 2nd. ed. NY: Oxford University Press Inc., 1988.

UNESCO. Background papers and supporting data on the international equation of state of seawater 1980. *UNESCO Technical papers in marine science*, n. 38, 1981.

WILCOX, D. *Turbulence modeling for CFD*. La Cañada, CA: DCW Industries, 1993.

WRIGHT, N. Introduction to numerical methods for fluid flow. In: BATES, P. D.; N.LANE, S.; FERGUSON, R. I. (Ed.). *Computational Fluid Dynamics: Applications in Environmental Hydraulics*. England: John Wiley & Sons Ltd, 2005. cap. 7, p. 147–68.

ZIENKIEWICZ, O.; TAYLOR, R. *The finite element method: Volume 1: The basis*. Ma: Butterworth-Heinemann, 2000.

ZIENKIEWICZ, O.; TAYLOR, R. *The finite element method: Volume 3: Fluid dynamics*. Ma: Butterworth-Heinemann, 2000.