

**Rafael Antonio Sanabria Villalobos**

**Object-Based Modeling of Fluvial Reservoirs**

**DISSERTAÇÃO DE MESTRADO**

Dissertation presented to the Programa de Pós-graduação em Matemática of the Departamento de Matemática PUC-Rio as partial fulfillment of the requirements for the degree of Mestre em Matemática.

Advisor: Prof. Sinesio Pesco

Rio de Janeiro  
April 2016

**Rafael Antonio Sanabria Villalobos**

## **Object-Based Modeling of Fluvial Reservoirs**

Dissertation presented to the Programa de Pós-graduação em Matemática of the Departamento de Matemática do Centro Técnico Científico da PUC-Rio, as partial fulfillment of the requirements for the degree of Mestre.

**Prof. Sinesio Pesco**

Advisor

Departamento de Matemática - PUC-Rio

**Prof. Lis Ingrid Roque Lopes Custódio**

Instituto Politécnico do Rio de Janeiro - UERJ

**Prof. Abelardo Borges Barreto Junior**

Departamento de Engenharia Mecânica - PUC-Rio

**Prof. Márcio da Silveira Carvalho**

Coordinator of the Centro Técnico Científico da PUC-Rio

Rio de Janeiro, April 7th, 2016

All Rights Reserved.

**Rafael Antonio Sanabria Villalobos**

Has a degree in Mathematics Education from Universidad de Costa Rica, Costa Rica.

Bibliographic data

Sanabria Villalobos, Rafael Antonio

Object-Based Modeling of Fluvial Reservoirs/ Rafael Antonio Sanabria Villalobos; advisor: Sinesio Pesco. — Rio de Janeiro: PUC-Rio, Matemática, 2016.

v., 61 f: il. ; 29,7 cm

1. Dissertação (mestrado) - Pontifícia Universidade Católica do Rio de Janeiro, Departamento de Matemática.

Inclui referências bibliográficas.

1. Matemática – Dissertação 2. Fluvial Reservoirs 3. Object-Based Facies Modelling 4. Simulated Annealing I. Pesco, Sinesio. II. Pontifícia Universidade Católica do Rio de Janeiro. Departamento de Matemática. III. Título.

CDD: 510

## Acknowledgement

I would like to express my sincere gratitude to CAPES for the aid granted.

My sincere thanks to PETROBRAS for the aid granted.

I would like to thank my advisor Prof. Sinesio Pesco for the continuous support, for his patience, motivation, and immense knowledge.

Besides my advisor, I would like to thank the rest of my thesis committee for their suggestions and contributions, and for accepting the invitation.

My sincere thanks also goes to Creuza and all administrative staff of the Department of Mathematics for helping me to fulfill all the bureaucracy requirements.

I would also like to thank my mother and sisters who have sacrificed a lot. We have worked together to move forward, thank you very much to all three and the whole family.

To my friends that are many and very important in my life.

To my grandfather who gave me everything.

I especially want to thank my wife for being the perfect accomplice in this craziness.

## Abstract

Sanabria Villalobos, Rafael Antonio; Pesco, Sinesio (Advisor). **Object-Based Modeling of Fluvial Reservoirs**. Rio de Janeiro, 2016. 61p. MSc. Dissertation — Departamento de Matemática, Pontifícia Universidade Católica do Rio de Janeiro.

The problem addressed in this work is to model fluvial reservoirs using geostatistical model based on objects. This work aims to create an object-based model for fluvial systems which simulates, given some observed data, two of the principal facies associated to the fluvial reservoirs: the channel sand and crevasse splay sand. Also, it applies a simulation with constraints based on the Simulated Annealing approach.

## Keywords

Fluvial Reservoirs; Object-based Facies Modelling; Simulated Annealing.

## Resumo

Sanabria Villalobos, Rafael Antonio; Pesco, Sinesio. **Modelagem de Reservatórios Fluviais Baseada em Objetos**. Rio de Janeiro, 2016. 61p. Dissertação de Mestrado — Departamento de Matemática, Pontifícia Universidade Católica do Rio de Janeiro.

O problema tratado nesta dissertação consiste em modelar reservatórios fluviais utilizando modelos geoestatísticos baseados em objetos. Este trabalho propõe um modelo baseado em objetos para sistemas fluviais que simula, conforme dados observados, duas das principais faces associadas aos reservatórios fluviais: o canal e o espraçamento de crevasse. O modelo proposto realiza uma simulação com condicionantes baseado na técnica Simulated Annealing.

## Palavras-chave

Reservatórios Fluviais; Modelo de faces baseado em objetos; Simulated Annealing.

# Contents

1	Introduction	<b>9</b>
1.1	Literature Review	10
1.2	Dissertation Outline	11
2	Basic Concepts	<b>12</b>
2.1	Some Geological Concepts	12
2.2	Geostatistical Modeling Terminology	15
3	Fundamental Modeling	<b>23</b>
3.1	Object Modeling	23
3.2	Simulating Model	30
4	Implementation Results	<b>43</b>
4.1	Objects Construction	43
4.2	Simulations	46
5	Conclusion	<b>57</b>
	Bibliography	<b>59</b>
	Index	<b>61</b>

## List of Figures

2.1	Cross section of a Reservoir	13
2.2	Formation of a crevasse splay in the Columbia River, British Columbia, Canada (picture by H.J.A. Berendsen). From <a href="http://www.geo.uu.nl/fg/palaeogeography">http://www.geo.uu.nl/fg/palaeogeography</a>	14
2.3	B-Spline with five control points	17
2.4	Probability of accepting a change to the system in simulated annealing	20
2.5	Simulated annealing process	22
3.1	Channel with sinuosity $s = 1$ .	24
3.2	Channel with sinuosity $s_1 = s_2 = \pi/2$ .	25
3.3	Construction of channel center line.	25
3.4	Construction of channel's boundary.	26
3.5	Cross sectional view of channel construction.	26
3.6	Channel's positioning.	26
3.7	Construction of the crevasse in the $xy$ plane.	28
3.8	Construction of the crevasse in the $yz$ plane.	29
3.9	Crevasse's Rotation.	30
3.10	Well data	31
3.11	Examples of planar grids.	33
3.12	Examples of 3-D grids.	33
3.13	Cross section view of the grid in case of channels' intersection.	34
3.14	Channel in-out function	35
3.15	Channel grid 3-D matrix.	36
3.16	The condition is no longer true, so the model calculates the best $y$ .	37
3.17	The channel is translated to honor the well data.	38
3.18	Modeling process	41
3.19	Filling wells process	42
4.1	Construction of the crevasse	44
4.2	Crevasse in-out function	44
4.3	Construction of a channel	45
4.4	Construction of a channel with its respective crevasses	45
4.5	Channel with width $\omega = 2$ and thickness $t = 0.75$	46
4.6	Channel with width $\omega = 1$ and thickness $t = 1.8$	46
4.7	Channel honoring well data	47
4.8	Simulation with one well.	48
4.9	Simulation with three wells	49
4.10	Three wells and more crevasses	50
4.11	Global proportion of 10%	52
4.12	Global proportion of 30%	54
4.13	Simulation with two wells and 6.5% global proportion	56



# 1

## Introduction

A fluvial reservoir is a sedimentary deposit consisting of material transported by, suspended in, or laid down by a stream [Neuendorf, 2005]. It is characterized by the presence of sinuous channels. They may be the most important hydrocarbon reservoirs in the world. Hence, reservoir characterization has become increasingly important for estimating reserves as the demand for hydrocarbon resources continues to grow.

Modeling this type of reservoirs is challenging because they are physically inaccessible. The reservoir is located underground and the channels can only be described through indirect measurements. This makes it difficult to define the channel parameters.

A well is a borehole which is drilled in order to discover or delimit a petroleum deposit. The information provided by each perforation is called well data (see figure 3.10(a)), and provides precise pointwise locations of the channels. However, in the interwell areas the channels are positioned arbitrarily, which makes it difficult to determine their spatial distribution between wells. Therefore, it is necessary to create the geological evolution of the basin of the reservoir with the few measured data.

Modeling heterogeneous reservoirs is classically performed using geostatistical methods. Those methods reproduce the spatial distribution of heterogeneities based on available data.

The objective of this work is to estimate the channels geometry and their spatial distribution, conditioned to the observed data. The approach adopted here is a hierarchical object-based modeling, combined with an optimization method. An object based model focuses in the construction of objects using class diagrams, the relationship between these classes, methods in the

classes and properties. On the other hand, the optimization method aims to solve the problem of approximating the global optimum of a given function.

## 1.1

### Literature Review

Fluvial reservoirs have been studied extensively, and several researchers have addressed the problem of characterizing them using geological intuition. Quantitative computer models for fluvial systems date back to at least 1978 [Allen, 1978]. In 1984 Johnson and Krol [Johnson et al., 1984] proposed hand-made models which rely on the geological interpretation of the well data, and were not able to represent the heterogeneities properly.

Alternative methods made use of stochastic simulations to describe the complex geological architecture honoring the well data and conditioned to probability distributions obtained from studies of analog depositional environments [Haldorsen and Lake, 1982]. Although the simulated models honored the observed data, they still incurred large uncertainty in the interwell areas.

Techniques developed in the 1990s involved the object-based approach for characterizing interwell heterogeneities. In 1995 researchers from Matmídia Laboratory of the Mathematical Department of Pontifícia Universidade Católica do Rio de Janeiro (PUC-Rio), developed a software for stochastic modeling of geological objects called **PetBool** [Barreto et al., 1995]. This software allows conditioned simulations for a great number of wells and proportion curves. It deals with the 3D geometry and heterogeneity of oil bearing reservoirs.

Poletto [Poletto, 1996] proposed a geometric modeling and stochastic simulation of fluvial reservoirs from the genetic channel unit. He did conditional boolean modeling based on subsurface data and using PetBool software. The author reported that “*The use of the parametric and implicit representations of the semi-ellipse makes the algorithm efficient to generate and to visualize objects. The volumes estimation associated to surfaces generated with this strategy is equally facilitated.*”

Deutsch and Wang [Deutsch and Wang, 1996] proposed in 1996 a systematic technique that allows the description of the geological features in fluvial system to be even more efficient and realistic. This technique is based

on a hierarchical approach which involves relative stratigraphic coordinates, translations, rotations, and straightening functions. Recently Deutsch and Tran [Deutsch and Tran, 2002] in 2002 created a computer code called **FLU-VSIM** which is also a hierarchical object-based modeling with some significant refinements like the use of simulated annealing and non-random perturbation rules for conditioning to extensive well data.

## 1.2

### Dissertation Outline

This dissertation is organized in four chapters. Chapter 2 presents the basic geological concepts necessary for the reader to understand the reservoir model presented in chapter 3. It also covers the principal mathematical tools used to construct the objects and the optimization method implemented.

Chapter 3 contains a detailed explanation of the object-based model developed here. It covers the objects construction, models constraints and simulation process. Chapter 4 shows the simulations created by the model, and chapter 5 presents a summary of the principal results obtained in this work.

## 2

## Basic Concepts

In this chapter we present some fundamental concepts related to geology and computational modeling. The first section presents the geological terminology employed in this work. The second section covers the computational tools used in the modeling process, this includes B-Splines curves and optimization methods.

### 2.1

#### Some Geological Concepts

We want to model fluvial reservoirs, which are subsurface pools of hydrocarbons contained in porous or fractured rock formations. The composition of every fluvial deposit is determined by the geology of the source areas from which the sediments were derived [Miall, 2006].

According to Neuendorf [Neuendorf, 2005] a **channel** is a natural, or human made, relatively shallow and narrow passageway or depression of perceptible extent containing continuously or periodically flowing water or any body of fluid. Most commonly it refers to the confine of a river, river delta or strait. Our main interest is to model the channels of this type of reservoir.

Natural channels are formed by fluvial process mostly formed by flowing water, although can also be formed by other fluids such as flowing lava.

Next to a channel there could be a **flood plain**. It refers to an area of flat land adjacent to a stream or river where the elevation of the land is lower than other areas, so it is subject to periodic flooding by the stream. It is produced by lateral movement of a stream. For example, when a river breaks its banks and floods, leaves unconsolidated sedimentary deposits or alluvium [Ezekiel et al., 2013]. These gradually build up to create the floor of the flood

plain.

As rivers rise over their banks the water leaves the normal stream channel and enters the flood plain. The current slows at the water's edge and part of the sediment load in the water is dropped on the banks. Over a significant number of floods this builds up a gentle rise which is called **levee**. They could also form underwater, in submarine canyons by the same mechanism as river levees.

With every channel there will also appear several crevasses. The term **crevasse** refers to a deep crack or fracture found in the earth surface, usually in an ice sheet or glacier. In the present work it refers to a wide breach or crack in the bank of a channel (see Figure 2.1).

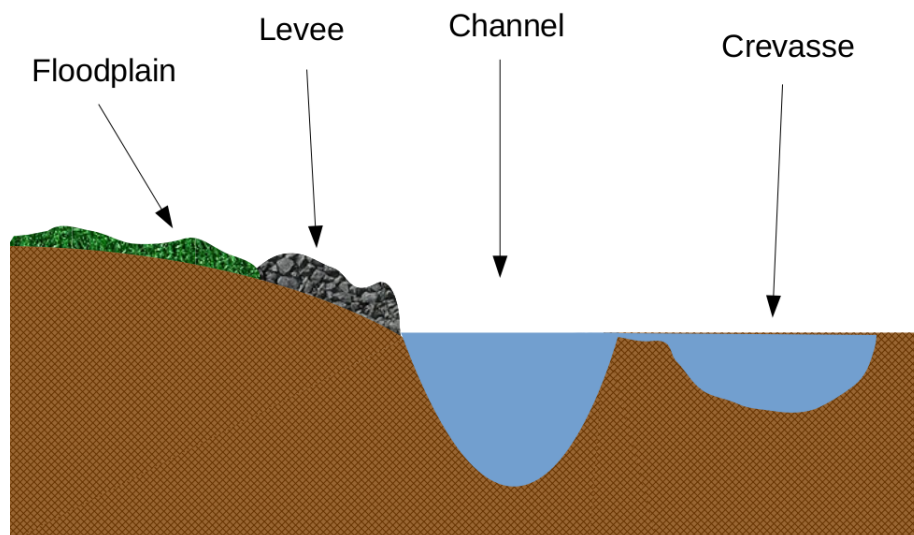


Figure 2.1: Cross section of a Reservoir

When a stream, during excess discharges, breaks its natural or artificial levees, and deposits sediment on a flood plain it forms a crevasse splay. According with Neuendorf [Neuendorf, 2005] a crevasse splay is “a low energy delta, oriented perpendicular to a main channel, formed by a break in the natural levee during flood stage.”

Those breaches occur most commonly on the outside banks of meanders, where the water stream has the highest energy. That is, the crevasse splay mostly appear after a extreme U-bend in the course of a stream (see Figure 2.2).



Figure 2.2: Formation of a crevasse splay in the Columbia River, British Columbia, Canada (picture by H.J.A. Berendsen). From <http://www.geo.uu.nl/fg/palaeogeography>

Object-based modeling becomes important due to the geometric features associated with the main facies presented in such depositional environments. A **depositional environment** is any area where sediment is capable of accumulating [Tvelin, 2016]. Different depositional environment will have different structure and texture of sediments. The specific kinds of deposits are determined by the physical, chemical, and biological characteristics of the depositional environment.

According to Hall [Hall, 2007] “a **facies** is the aspect, appearance, and characteristics of a rock unit, and it usually reflects the conditions of the origin of the unit.” Those characteristics reflect its original composition as well as its environment of deposition, and allow it to be distinguished from rock or sediment deposited in an adjacent environment. Every depositional environment puts its own distinctive imprint on the sediment, making a particular facies called sedimentary facies.

Sedimentary facies reflect depositional environment, each facies being a distinct kind of sediment for that area or environment. A sedimentary facies could comprise different sorts of deposits, the difference between them may be its composition, texture, bedding, organic content, or any combination of these. They may be developed at a given place only in the lower part, the middle part, or the upper part of strata. Thus, rocks belonging to one facies may underlie, overlie or be intercalated between rocks of another facies [Longwell, 1949].

Sedimentary rocks can be formed only where sediments are deposited long enough to become compacted and cemented into hard beds or strata.

The process of sedimentation commonly occurs in areas where the sediment lies undisturbed for many years in sedimentary basins.

Generally, facies are distinguished by what aspect of the rock or sediment is being studied. Thus, facies based on petrological characters such as grain size and mineralogy are called lithofacies, whereas facies based on fossil content are called biofacies.

For the purpose of this work, there are two facies types. The first type is **channel sand**, which has high energy of deposition and consequent coarser grain size. The second facies type is **crevasse splay sand** which is the sand formed when the levee is breached and it is deposited away from the main channel.

## 2.2

### Geostatistical Modeling Terminology

#### 2.2.1

##### B-Splines

The first step to follow in order to model a fluvial reservoir is modeling the channel. The sinuous channel shapes are modeled by a 1-D centerline, which is determined by control points<sup>1</sup>.

There are several ways to represent the curve determined by those control points, for example by explicit or implicit form (see [Piegl and Tiller, 1997]). However, the parametric form is more natural for designing and representing shape in a computer. Given  $n + 1$  control points  $P_0, \dots, P_n$ , the objective is to find the curve  $C(t)$  determined by them. One first approximation could be to use *Bezier curves*<sup>2</sup> which are given by the equation

$$C(t) = \sum_{j=0}^n B_j^n(t) P_j, \quad (2.1)$$

where  $t$  is the parameter taken in the interval  $[0, 1]$  and

$$B_j^n(t) = \frac{n!}{j!(n-j)!} t^j (1-t)^{n-j}. \quad (2.2)$$

<sup>1</sup>For the purpose of this work those points would be randomly taken

<sup>2</sup>For a more detail study of Bezier curves see [Piegl and Tiller, 1997]

However, locally the curve cannot easily be controlled. That is, any change to an individual control point will cause changes in the curve along its full length. Another disadvantage of this approach is that any additional point will automatically increase the degree of the curve, so it is not possible to keep the degree of the Bezier curve fixed while adding additional points.

The **B-Spline** curve addresses each of these problems (see Figure 2.3). Let  $T = \{t_0, \dots, t_m\}$  be a nondecreasing sequence of real numbers, i.e.  $t_j \leq t_{j+1}$  for all  $j = 0, \dots, m-1$  where  $m = n + k$  and  $n \geq k-1$ . The  $t_j$  are called *knots* and  $T$  is called the *vector knot* or *knot sequence*. The B-Spline is defined for  $t \in [t_0, t_m]$  as

$$C(t) = \sum_{j=0}^n N_{j,k}(t) P_j. \quad (2.3)$$

Where the  $N_{j,k}(t)$  is the  $j$ th B-Spline basis function of degree  $k$ , that is defined by

$$N_{j,0}(t) = \begin{cases} 1, & t_j \leq t \leq t_{j+1}, \\ 0, & \text{otherwise.} \end{cases} \quad (2.4)$$

and

$$N_{j,k}(t) = \frac{t - t_j}{t_{j+k} - t_j} N_{j,k-1}(t) + \frac{t_{j+k+1} - t}{t_{j+k+1} - t_{j+1}} N_{j+1,k-1}(t). \quad (2.5)$$

Note that the curve parameter,  $t$ , is not restricted to a unit range as in the Bezier curve.

The B-Spline curve has the following properties (see [Piegl and Tiller, 1997]):

1.  $N_{j,k}(t) = 0$  if  $t$  is outside the interval  $[t_j, t_{j+k+1})$ .
2. In any given knot span  $[t_j, t_{j+1})$ , at most  $k+1$  of the  $N_{j,k}$  are nonzero, namely the functions  $N_{j-k,k}, \dots, N_{j,k}$ .
3. Nonnegativity:  $N_{j,k}(t) \geq 0$  for all  $j, k$ , and  $t$ .
4. Partition of unity: for an arbitrary knot span  $[t_j, t_{j+1})$  and for all  $t \in [t_j, t_{j+1})$

$$\sum_{i=j-k}^j N_{i,k}(t) = 1. \quad (2.6)$$



5. Geometry invariance property: the partition of unity property assures the invariance of the shape of the B-Spline curve under translation and rotation.
6. All derivatives of  $N_{j,k}(t)$  are defined in the interior of a knot span. At a knot  $N_{j,k}(t)$  is  $k - l$  times continuously differentiable, where  $l$  is the multiplicity of the knot.
7. Except for the case  $k = 0$ ,  $N_{j,k}(t)$  attains exactly one maximum value.
8. Convex hull property: the curve is contained in the convex hull of its control polygon.
9. Local modification scheme: changing the point  $P_j$  only changes  $C(t)$  in the interval  $[t_j, t_{j+k+1})$ .
10. The control polygon represents a piecewise linear approximation to the curve. This approximation is improved by knot insertion or degree elevation.

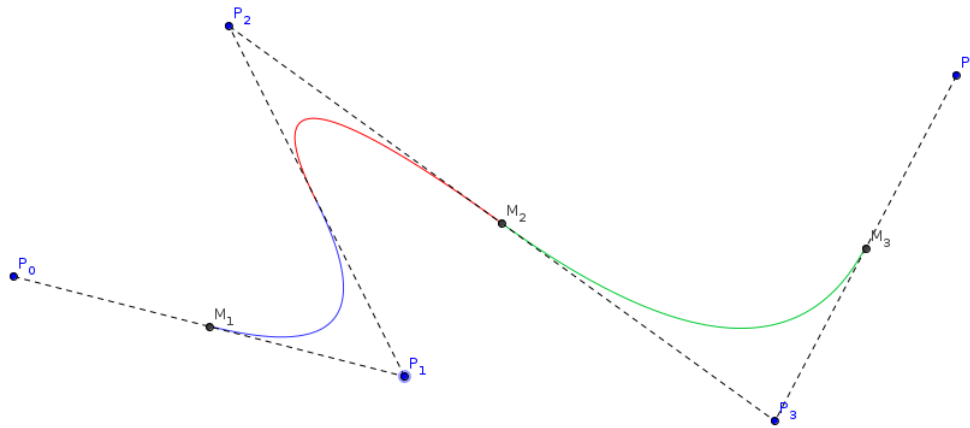


Figure 2.3: B-Spline with five control points

Due the properties above mentioned, we opt for use the B-Splines to model the Chanel, as well the crevasses and the crevasse's depth that will be the subject of the next chapter.

## Nonparametric B-Splines curves

A **nonparametric** B-Spline curve is a planar nonrational B-Spline curve that is used to approximate a single-valued curve  $y = y(x)$  in Cartesian coordinates  $(x, y)$ .

The curve  $C(t)$  of degree  $k$  is defined over  $n$  intervals by the vector knot  $\{t_0, \dots, t_{n+2k-2}\}$ . Its domain is restricted to  $t \in [t_{k-1}, t_{n+k-1}]$ . The nonparametric form of the curve is based on the form of its control points namely  $P_j = (x_j, p_j)$  where

$$x_j = \frac{1}{k}(t_j + \dots + t_{j+k-1}), \quad j = 0, \dots, n+k-1. \quad (2.7)$$

This ensures that the curve  $C(t)$  has the following form:

$$C(t) = \begin{bmatrix} x(t) \\ y(t) \end{bmatrix} = \begin{bmatrix} t \\ f(t) \end{bmatrix}, \quad (2.8)$$

where the second coordinate has the single-valued expression:

$$f(t) = \sum_{j=0}^{n+k-1} p_j N_{j,k}(t), \quad (2.9)$$

with  $N_{j,k}$  being the  $j$ th B-Spline basis function as before. For a more detailed study in nonparametric B-Spline curves see [Farin, 1996].

For the rest of this work, it will be assumed that each B-Spline curve is a nonparametric curve.

### 2.2.2

#### Simulated Annealing

Another approach to construct reservoir models came from optimization. The optimization methods are able to impose additional constraints that either can be incorporated into an objective function or may be coupled to a forward model. Optimization for model construction includes complicated data, statistical constraints and model updating to impose these constraints.

There are many optimization algorithms, including hill climbing, genetic algorithms, gradient descent, and more (see [Hopgood, 2001]). Simulated annealing's strength is that it avoids getting caught at local maxima. It decides

when to accept or reject a given perturbation.

The simulated annealing algorithm, first introduced by Kirkpatrick in 1982 [Kirkpatrick et al., 1983], was originally inspired from the process of annealing in metal work as referred in table 2.1. Annealing is a process in metallurgy where metals are heated and slowly cooled under controlled conditions to alter its physical properties. These alterations are due to the changes in its internal structure which make them reach a state where they are very strong. The heat increases the energy of the atoms allowing them to move freely and, as the temperature is slowly lowered, the molecules line up in crystals which represent the minimum energy state for the system.

Annealing	Simulated Annealing
Initial High Temperature	$T_i$
Final Low Temperature	$T_f$
Controlled Cooling Process	$f(t)$
System States	Feasible Solutions $x$
	Objective function $O(x)$

Table 2.1: Physical Annealing & Simulated Annealing

In simulated annealing, the temperature (control parameter  $t$ ) is kept variable to simulate this heating process. Heating the system results in a relaxation of the acceptance criteria of the samples taken from the search space. As the system is cooled, the acceptance criteria of samples, determined by a probability, is narrowed to focus on improving movements.

At each step, it considers some neighbouring state  $O_{new}$  of the current state  $O_{old}$ , and probabilistically decides between moving the system to state  $O_{new}$  or staying in state  $O_{old}$ . The acceptance probability distribution is given by:

$$P_{accept} = \begin{cases} 1, & O_{new} \leq O_{old}, \\ \exp\left(\frac{O_{old} - O_{new}}{t}\right), & \text{otherwise.} \end{cases} \quad (2.10)$$

The first condition checks if the neighbour solution is better than the current solution. If it is, the algorithm accepts it unconditionally. If however, the neighbour solution is not better it is necessary to consider a couple of factors. Firstly, how much worse the neighbour solution is; and secondly, how high is the current “temperature” of the system. At high temperatures the

system is more likely to accept solutions that are worse, second condition (see Figure 2.4).

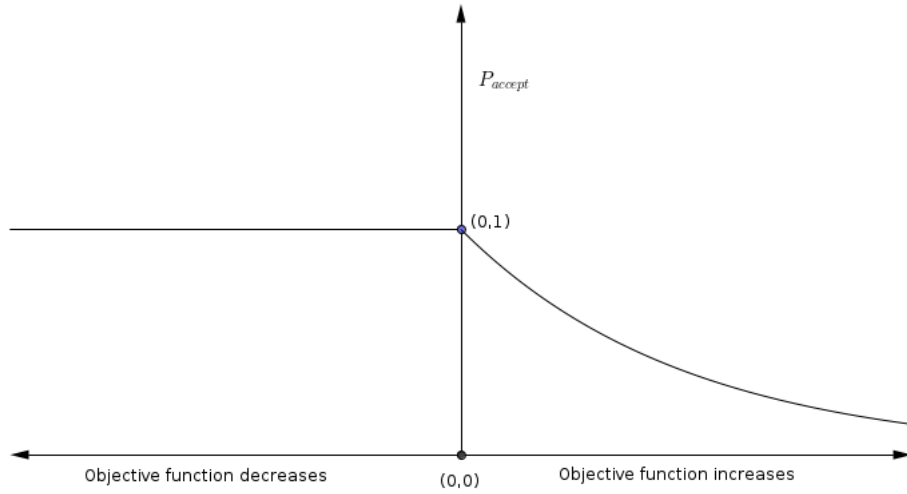


Figure 2.4: Probability of accepting a change to the system in simulated annealing

Simulated annealing requires an *annealing schedule*, which specifies how the temperature is reduced as the search progresses. It must not be lowered too fast or else the image may get trapped in a suboptimal situation and never converge. However, if lowered too slowly, then convergence may be unnecessarily slow.

Geometric cooling is one of the most widely used schedules, where  $f(t) = t\alpha$  where  $\alpha < 1$ . The probability that the simulated annealing algorithm terminates with a global optimal solution approaches 1 as the annealing schedule is extended.

The following pseudo-code exemplifies how simulated annealing works. Here  $T_i$  and  $T_f$  mean initial and final temperature, respectively. The starting temperature must be hot enough (close to 1), to allow a move to almost any neighbour state. However, if the temperature starts too high then the search can move to any neighbour state and thus it transforms (at least in the early stages) into a random search.

The stopping criteria can either be a suitably low temperature (close to zero) or when the system is “frozen” at the current temperature, i.e. no

better or worse moves are being accepted.

---

**Algorithm 1** Simulating Annealing Algorithm
 

---

```

1: CONSTRUCT { initial solution  $x_0$ ;  $O_{old} = x_0$ };
2: SET { initial temperature  $t = T_i$ };
3: while  $t > T_l$  do
4:   generate randomly a neighbouring solution  $O_{new} \in N(O_{old})$ ;
5:   compute change of cost  $\Delta O = O_{new} - O_{old}$ ;
6:   if  $\Delta O \leq 0$  then
7:      $O_{old} = O_{new}$ 
8:   else
9:     generate  $q = \text{Random}(0, 1)$ ;
10:    if  $q < \exp\left(\frac{\Delta O}{t}\right)$  then
11:       $O_{old} = O_{new}$ 
12:    end if
13:  end if
14:  SET {new temperature  $t = f(t)$ };
15: end while
16: RETURN { solution corresponding to the minimum objective function }

```

---

Simulated annealing requires an *objective function* that is a suitable measure of the discrepancy between the data and the model. This technique bases its operation in considering the objective function of the minimization problem as the equivalent energy function of an illusory annealing process.

There are, in general, multiple data sources that are coded as separated component objective functions. The total objective function is formed by a weighted sum of these components:

$$O = \sum_{i=1}^{N_c} \omega_i O_i, \quad (2.11)$$

where  $N_c$  is the number of components, and for each  $i = 1, \dots, N_c$ ,  $\omega_i$  is the weight, and  $O_i$  is the component objective function.

Every component objective function must be positive and decrease to zero when data or spatial statistics are reproduced by the realization. The rate at which each component objective function decreases to zero is different.

After every perturbation each component must be updated or recalculated. Global recalculation of the total objective function is avoided due to CPU cost. The general simulated annealing algorithm is presented in 1 and the process is summarize in the flux chart presented in 2.5.

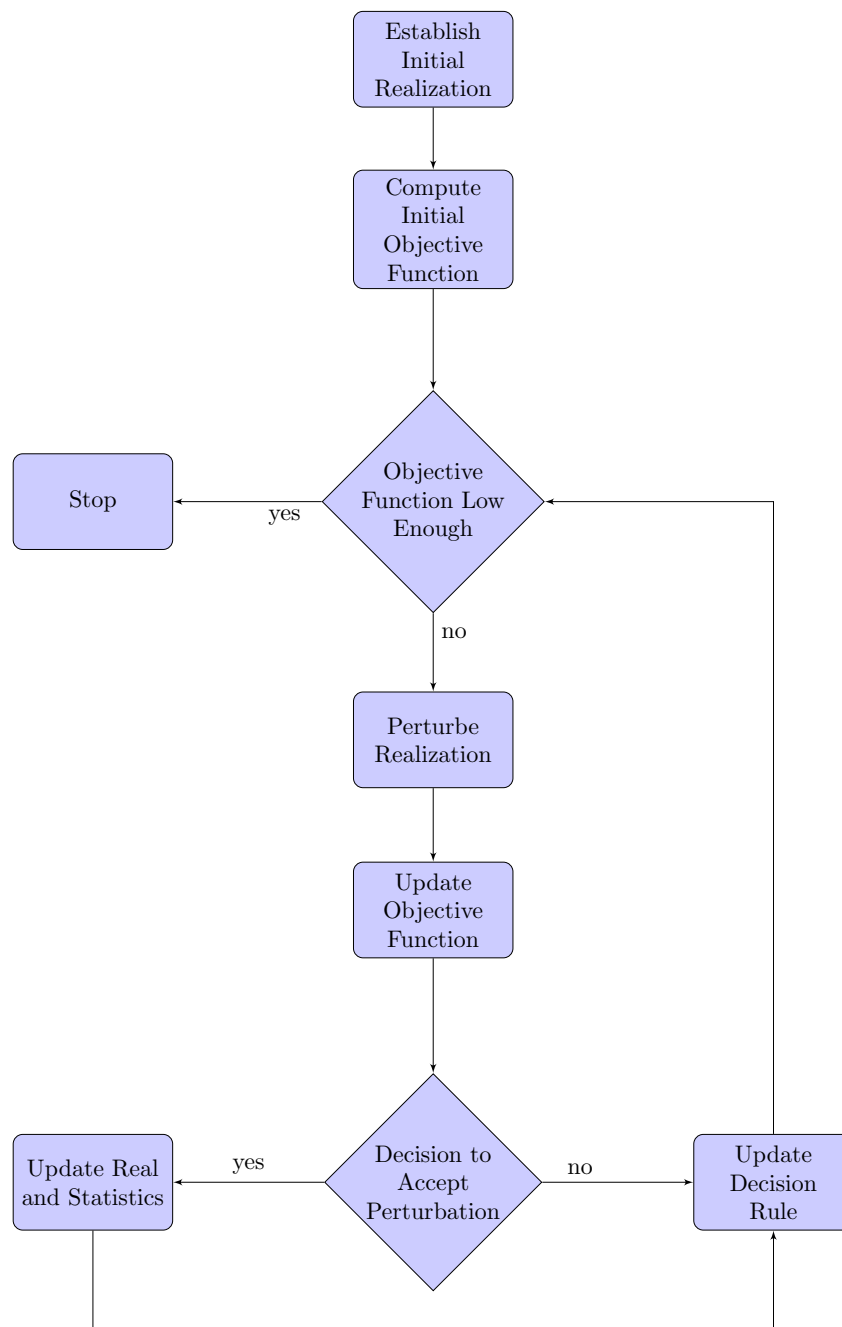


Figure 2.5: Simulated annealing process

### 3

## Fundamental Modeling

This chapter is dedicated to explaining the model constructed in this work. This model is inspired by FLUVSIM, an object-based program presented by Deutsch and Tran [Deutsch and Tran, 2002], for stochastic modeling of fluvial depositional systems. The FLUVSIM focuses on four facies types, but in the present work only two of them are studied. Each of these facies has its own geometric specification chosen to mimic the shapes idealized from observation.

The *Object Modeling* section covers the construction of the objects, the parameters that define every object and the tools used to construct them. The *Simulation Model* section explains how the model works. First it explains the two constraints that can be applied to the model, the grid and how the objects are labeled in it. Finally, this section covers the simulation process and how it honors the constraints.

### 3.1

#### Object Modeling

The first facies type that we are going to model is the **channel sand** of the fluvial reservoir. “This facies is viewed as the best reservoir quality due to the relatively high energy of deposition and consequent coarser grain size” [Deutsch and Tran, 2002]. The channels filled by this facies are geometrically defined by the channel width and thickness.

Next we turn our attention to the **crevasse splay sand**. As mention in the previous chapter, it is formed during excess discharges, when the stream breaks its natural levees, and deposits sediments on a flood plain.

### 3.1.1

#### Channel Object

The physical system under study is a fluvial reservoir. In this type of physical systems the dominant objects are channels with sinuous shapes. The geometry of the “object” **channel** is defined by the following parameters:

- *Channel center*: is a fixed point with rectangular coordinates  $(x, y, z)$ , which defines the channel’s position.
- *Sinuosity*: The sinuosity of a channel is its tendency to move back and forth across its floodplain, in an S-shaped pattern, over time. It is defined by 16 parameters namely  $s_i$ ,  $i = 1, \dots, 16$ .
- *Width* (parameter  $\omega$ ): it determines how wide the channel is.
- *Thickness* (parameter  $t$ ): it determines the maximum depth of the channel.
- *Direction* (parameter  $\alpha$ ): it determines the rotation of the channel on the  $xy$  plane. The rotation is made counter clockwise from the positive  $x$ -axis.

The amplitude  $l$  of the channel is determined by its sinuosity as follows (Poletto, 1996):

$$l = \Delta x \cdot \frac{2s - 2}{\pi - 2}, \quad (3.1)$$

where  $s$  is its sinuosity, and  $\Delta x$  is the constant of segmentation of the channels directrix<sup>1</sup>. The constant of segmentation is chosen as

$$\Delta x = \frac{2d}{16}, \quad (3.2)$$

where  $d$  is the diagonal of the area of interest. The sinuosity parameter is such that if  $s = 1$ , then we obtain a straight line and if  $s = \pi/2$ , then  $l = \Delta x$  (see Figures 3.1 and 3.2).

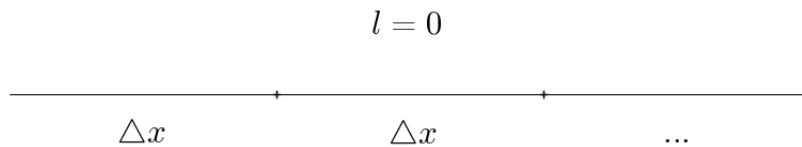
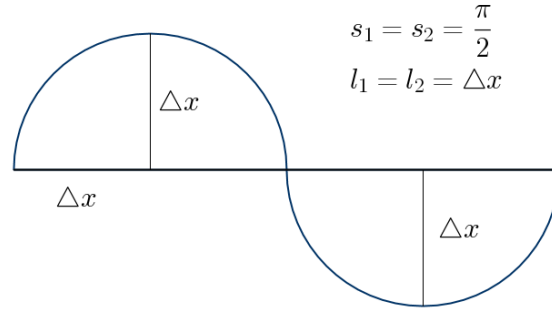


Figure 3.1: Channel with sinuosity  $s = 1$ .

<sup>1</sup>The channel directrix will be a segment of the  $x$ -axis.



Figure 3.2: Channel with sinuosity  $s_1 = s_2 = \pi/2$ .

According to Pyrcz and Deutsch [Pyrcz and Deutsch, 2014] “The sinuous channel shapes are modeled by a 1-D center line and a variable cross section along the center line”, and crevasse objects can be attached to the channels. The *channel center line* is a B-Spline curve defined by 16 control points (see Figure 3.3). Those control points are taken at every increment  $\Delta x$  on the directrix line, where a sinuosity value is chosen (given a distribution) to obtain the corresponding amplitude.

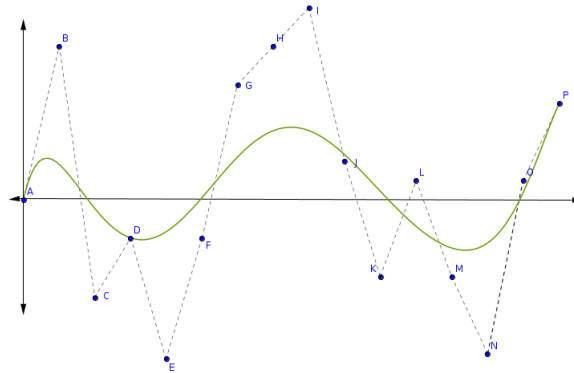


Figure 3.3: Construction of channel center line.

Figures 3.4 and 3.5 show the construction of the channel. Consider the cross section of the channel in the plane  $yz$ . There are four points determined by it. First, there is a point in the center line  $D$ . Then the two boundary points  $B_1$  and  $B_2$ , each one at a distance of  $\omega/2$  of  $D$ . Finally, the thickness point  $T$ , which is chosen as the point at distance  $t$  of  $D$  in the direction orthogonal to the segment defined by  $B_1$  and  $B_2$ . Now, it is possible to define the ellipse passing through the points  $B_1$ ,  $B_2$  and  $T$ . This half ellipse determines the form of the channel.

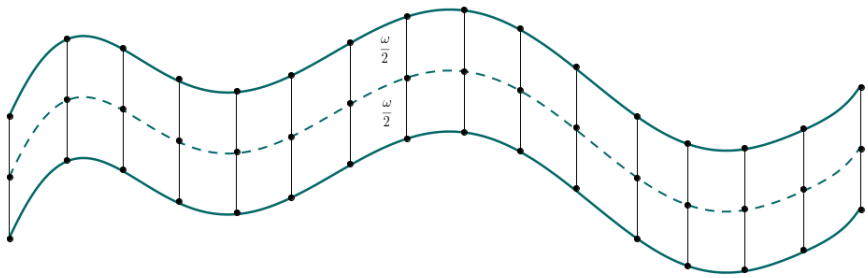


Figure 3.4: Construction of channel's boundary.

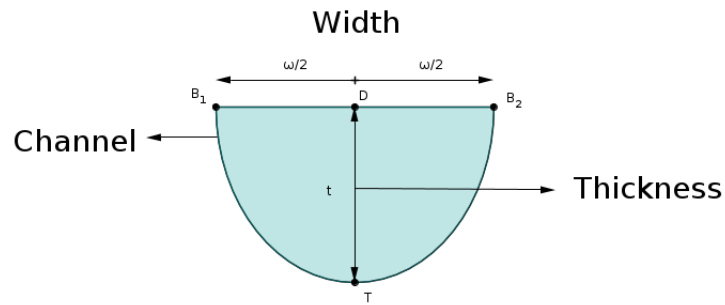


Figure 3.5: Cross sectional view of channel construction.

Finally, it is necessary to determine the position of the channel. In order to do this, we assume that the beginning of the center line is at the origin. Then translate it so its middle point coincides with the channel center  $(x,y,z)$ . After that, rotate the  $x$ -axis, and the channel with it, in the angle  $\alpha$  counter clockwise as shown in Figure 3.6.

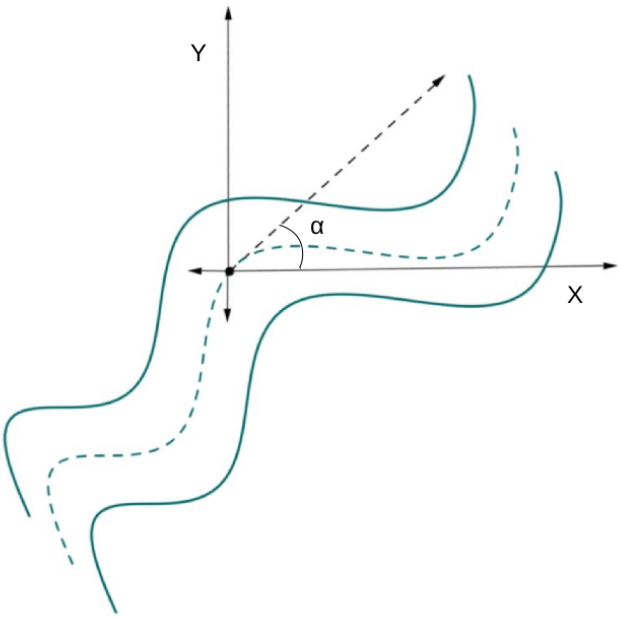


Figure 3.6: Channel's positioning.

### 3.1.2

#### Crevasse Object

The next object of the model presented here is the **crevasse**. This object modeling is inspired in the channel modeling, and the parameters defining it are:

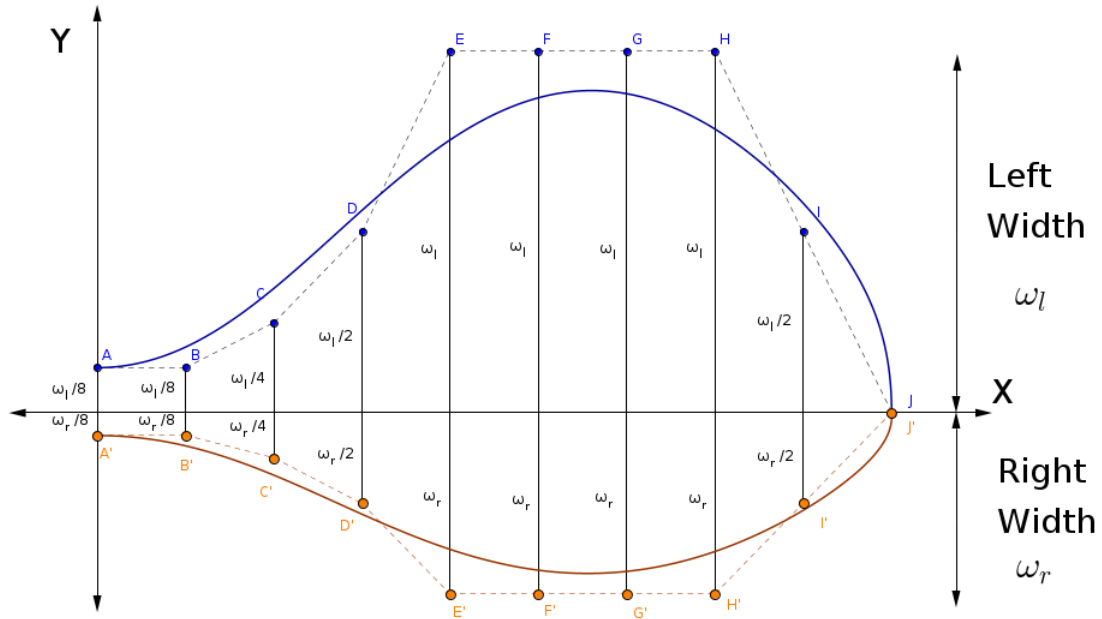
- *Length* (parameter  $l$ ): it determines the length of the crevasse.
- *Width*: it is defined by two parameters  $\omega_l$  and  $\omega_r$ ) that refer to the wide of the left hand side and the righthand side of the crevasse, respectively.
- *Thickness* (parameter  $t$ ): It determines the depth of the crevasse. This parameter is subordinated to the channel's thickness and the crevasse's length ( $l$ ). If the channel's thickness is less than  $l$ , then  $t$  equals the channel's thickness. If not,  $t$  will be half of the channel's thickness.
- *Center* (parameter  $x$ ): It determines the position of the crevasse in the channel. This parameter only depends on the  $x$  coordinate.
- *Direction* (parameter  $\beta$ ): it determines the rotation with respect to the crevasse's normal at the center point<sup>2</sup>.

For the construction of the crevasse in the  $xy$  plane, first assume that its based point  $O_c$  would be positioned at the origin of the plane. The crevasse's boundary in Figure 3.7 is defined by two B-splines curves given by the following control points:

Left Boundary $B_l$	Right Boundary $B_r$
$A = (0, \frac{\omega_l}{8})$	$A' = (0, \frac{\omega_r}{8})$
$B = (\frac{1}{9}l, \frac{\omega_l}{8})$	$B' = (\frac{1}{9}l, -\frac{\omega_r}{8})$
$C = (\frac{2}{9}l, \frac{\omega_l}{4})$	$C' = (\frac{2}{9}l, -\frac{\omega_r}{4})$
$D = (\frac{3}{9}l, \frac{\omega_l}{2})$	$D' = (\frac{3}{9}l, -\frac{\omega_r}{2})$
$E = (\frac{4}{9}l, \omega_l)$	$E' = (\frac{4}{9}l, -\omega_r)$
$F = (\frac{5}{9}l, \omega_l)$	$F' = (\frac{5}{9}l, -\omega_r)$
$G = (\frac{6}{9}l, \omega_l)$	$G' = (\frac{6}{9}l, -\omega_r)$
$H = (\frac{7}{9}l, \omega_l)$	$H' = (\frac{7}{9}l, -\omega_r)$
$I = (\frac{8}{9}l, \frac{\omega_l}{2})$	$I' = (\frac{8}{9}l, -\frac{\omega_r}{2})$
$J = (l, 0)$	$J' = (l, 0)$

Table 3.1: Points defining crevasse's boundary.

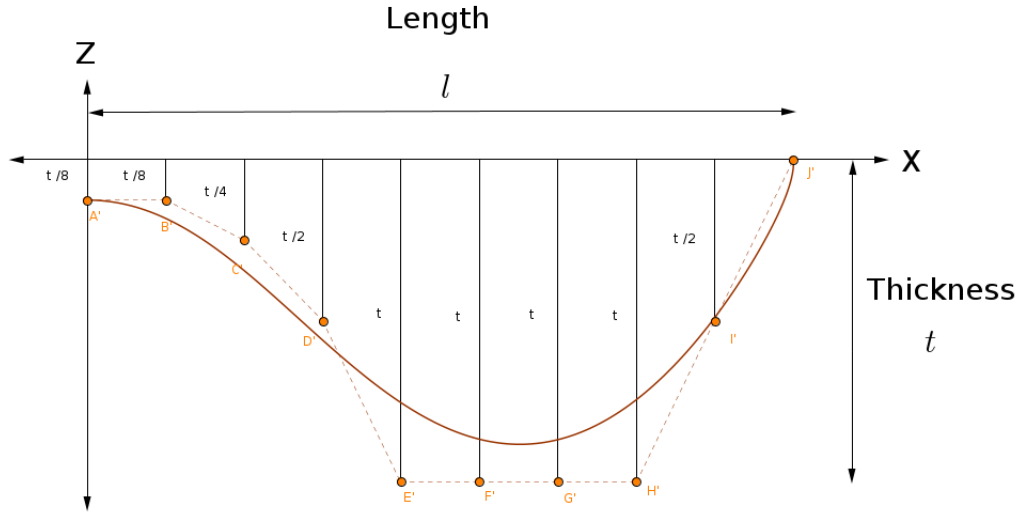
<sup>2</sup>In this work the angle  $\beta$  is chosen randomly.

Figure 3.7: Construction of the crevasse in the  $xy$  plane.

The next step is to construct the crevasse's depth. This will be done by defining another B-Spline curve whose control points will be given by the thickness of the crevasse and are presented in Table 3.2. This B-Spline curve will be placed at the  $yz$  plane as shown in Figure 3.8. The next step is to define, at each cross section, the ellipse determined by the boundary points in  $B_l$  and  $B_r$ , respectively, and the bottom point in  $T$ .

Depth Boundary $T$
$A' = (0, -\frac{t}{8})$
$B' = (\frac{1}{9}l, -\frac{t}{8})$
$C' = (\frac{2}{9}l, -\frac{t}{4})$
$D' = (\frac{3}{9}l, -\frac{t}{2})$
$E' = (\frac{4}{9}l, -t)$
$F' = (\frac{5}{9}l, -t)$
$G' = (\frac{6}{9}l, -t)$
$H' = (\frac{7}{9}l, -t)$
$I' = (\frac{8}{9}l, -\frac{t}{2})$
$J' = (l, 0)$

Table 3.2: Points defining crevasse's depth.

Figure 3.8: Construction of the crevasse in the  $yz$  plane.

The final step will be positioning it on the channel. Given the position parameter  $x$  it is possible to find its correspondent  $y$  on the channel center line<sup>3</sup>. Then one of the channel boundaries has to be chosen to locate the crevasse.

The crevasse is formed on the outside banks of meanders, so it should be placed in the convex side of the channel. To determine which side it is, we approximate the second derivative at the point  $(x, y)$  by finite differences. Once the approximation has been done we determine the sign of the second derivative. If it is positive, then the convex side will be the right one. If not, it will be the left one.

Now the crevasse should be translated. First the center  $O_c$  will be translated to the point  $(x, y)$ . Then we sum or subtract  $\omega/2$  depending on which boundary was chosen. Thus, the new center of the crevasse is:

$$O_c = \begin{cases} (x, y) + (0, -\frac{\omega}{2}), & \text{if second derivative positive,} \\ (x, y) + (0, \frac{\omega}{2}), & \text{otherwise.} \end{cases} \quad (3.3)$$

In order to rotate the crevasse consider the normal line at the point  $(x, y)$  and calculate its angle  $\theta$  with respect to the  $x$ -axis. Then rotate the center line of the crevasse by an angle  $\theta - \beta$  if the crevasse is on the left boundary of the channel, and by  $\theta + \beta$  if it is on the right boundary (see Figure 3.9). This

<sup>3</sup>This can be done because the B-Spline curve is given by its functional form.

last choice is based on the assumption that the stream's direction is from left to right.

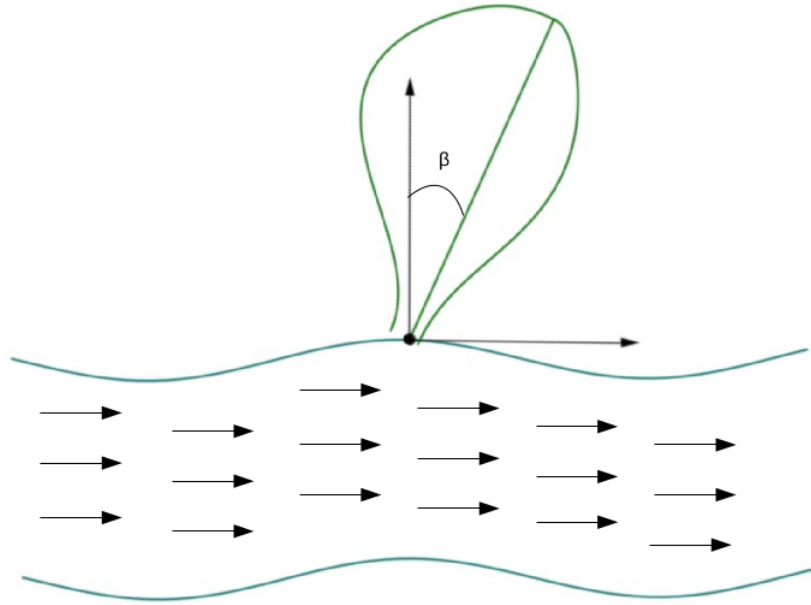


Figure 3.9: Crevasse's Rotation.

The model will determine the number of crevasses associated to a channel and its centers according to a distribution<sup>4</sup>.

It is important to mention that the crevasse's positioning should be done after the channel's construction but before the channel's positioning. This way the channel with its respective crevasses will be positioned together into the channel's position. After locating the crevasse in the channel, both the channel and the crevasse should be translated and rotated together with the channel parameters.

### 3.2 Simulating Model

Every reservoir model should be consistent with the available data. That observed data will generate a series of constraints that will give an idea about the reservoir's behaviour.

The *Constraints* section exhibits the observed data and statistical data that will conditionate the reservoir model. The *Grid* section explains the

<sup>4</sup>In this work we use an uniform distribution.

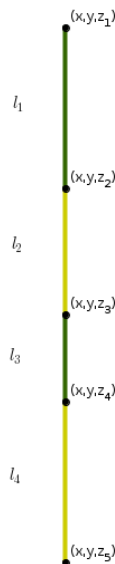
tools used to display the objects and to manipulate the constraints. The final section *Simulation Process* explains how the model uses the simulation mesh grid and constraints to simulate a reservoir model.

### 3.2.1 The Constraints

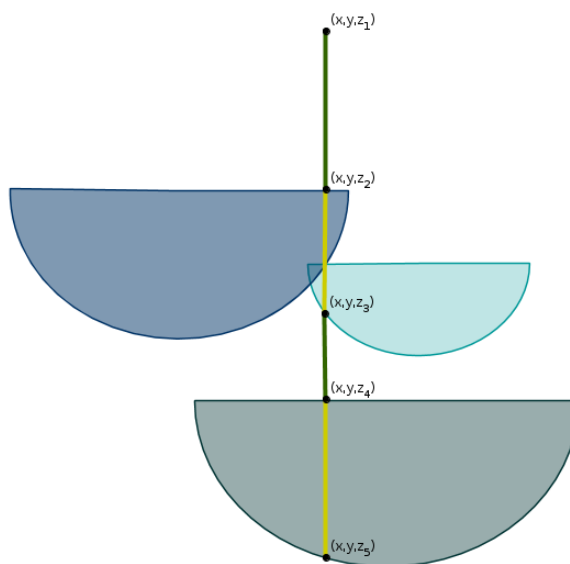
For the problem addressed here the first constraint is given by the **well data**. A well is a borehole which is drilled in order to discover or delimit a petroleum deposit. The information provided by each perforation is called well data (see Figure 3.10(a)), and specifies the following:

1. Well location: represented in rectangular coordinates as  $(x, y)$ .
2. Spatial distributions of channels: indicates the presence of a channel in specific depths of the well.

The spatial distribution data will be stored in two vectors. The first one will indicate the depths of the well, these depths will determine the partition of the well. The other one will be a vector filled with 0's and 1's which represents the indicator data. The indicator data represents which parts of the well will present channels.



3.10(a): Well.



3.10(b): Well filled by channels.

Figure 3.10: Well data

For example, the data in Figure 3.10(a) will be stored by the vectors  $I_1 = [z_1, z_2, z_3, z_4, z_5]$  and  $I_2 = [0, 1, 0, 1]$ . The first zero in  $I_2$  says that there should not be channels on the first interval indicated on  $I_1$ , that is  $[z_2, z_1]$ ; while the one in the second entrance of  $I_2$  means that there must be channels all the way through the second interval in  $I_1$ , that is  $[z_3, z_2]$ .

For now on, in every graphic representation of this work each interval that must contain channels will be represented in yellow, and the ones that do not will be presented in green. Also, in order to manipulate the well data, it will be stored in a matrix as shown in Table 3.3. To fill the matrix first enumerate the wells, and name the extreme values of  $z$  in every yellow interval as  $z_f$  for the maximum, and  $z_i$  for the minimum.

The first two columns of the matrix represent the  $x$  and  $y$  coordinate of the well. The third and fourth would be the extreme values of the  $z$  coordinate of each interval. The fifth column contains a condition that determines if the interval is filled by channels,  $-1$  means no and  $0$  means yes. The last column is the number of the well that contains the corresponding interval.

First Coordinate	Second Coordinate	Min $z$ $z_i$	Max $z$ $z_f$	Conditioning	Well number
$x$	$y$	$z_3$	$z_2$	$-1$	$0$
$x$	$y$	$z_5$	$z_4$	$-1$	$0$

Table 3.3: Matrix storing the data presented in Figure 3.10(a).

The other constraint is the **global proportion**. This is a parameter that determines the percentage of the volume that should be covered by channels and their respective crevasses. It is assumed that it will respect the well data, that means that it will be large enough to contain all the necessary channels to honored the well data.

### 3.2.2 Grid

The term *grid* usually refers to infinite sets of evenly spaced parallel lines at particular angles in a plane, or the intersections of such lines. The most common types of grid are orthogonal grids, and isometric grids such as the triangular grid (see Figure 3.11).



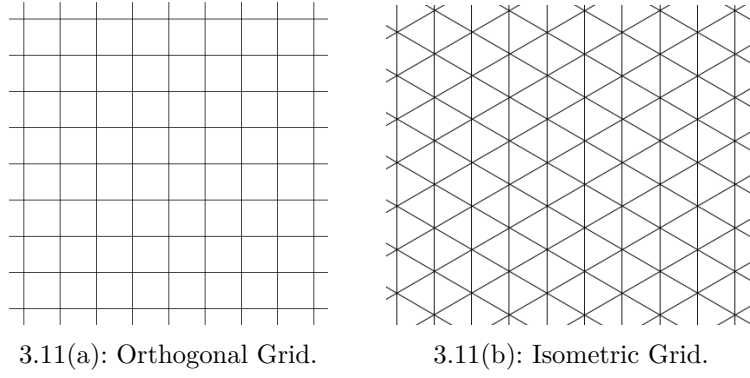


Figure 3.11: Examples of planar grids.

Grids can be generalized into  $n$ -dimensional space by using the centers of  $n$ -parallelotope as the points (see Figure 3.12). A **regular grid** in  $\mathbb{R}^3$  is a tessellation of the space by congruent parallelepipeds. A cartesian grid is a special case where the elements are unit cubes. There are other types of grids like the rectilinear grid whose elements are parallelepipeds that are not all congruent to each other, or the curvilinear grid where the cells are cuboids.

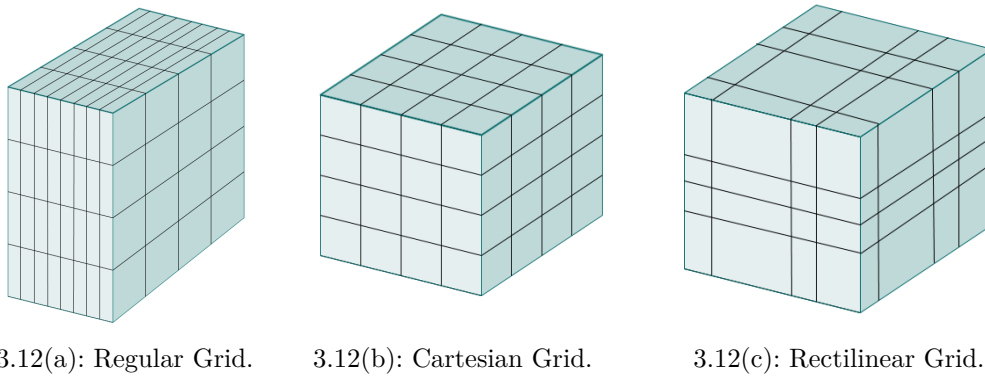


Figure 3.12: Examples of 3-D grids.

In this work we use a regular grid. The position of each parallelepiped is stored in a 3D matrix, i.e. a matrix of vectors.

During the model implementation, every cube in this grid (that is every matrix's entrance) will be labeled by a number, 1 means it is empty, 2 means that the cube is contained in a channel and 3 means that it belongs to a crevasse (see Figure 3.13).

If the same cube belong to different objects, it will be numbered by every one of them. For example, a cube numbered  $2^2 \cdot 3$  means it belongs to two channels and one crevasse.

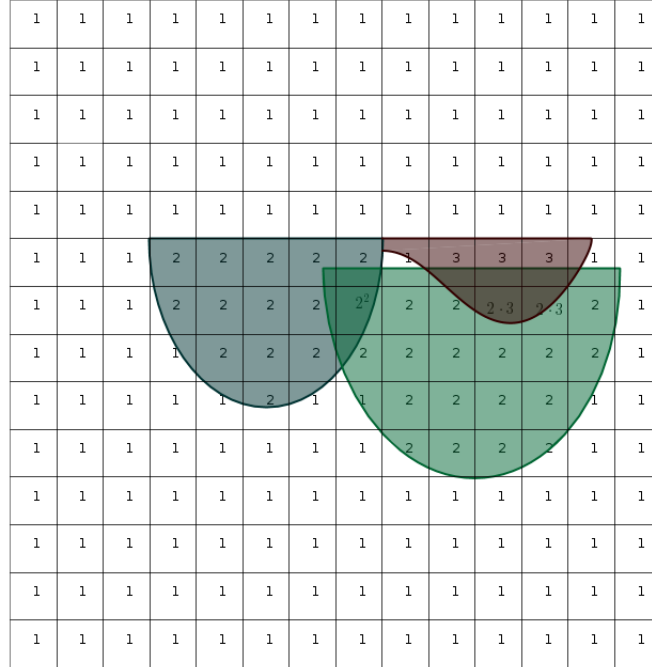
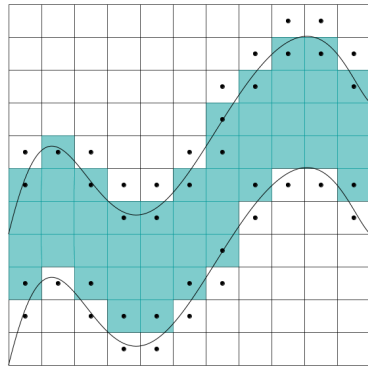


Figure 3.13: Cross section view of the grid in case of channels' intersection.

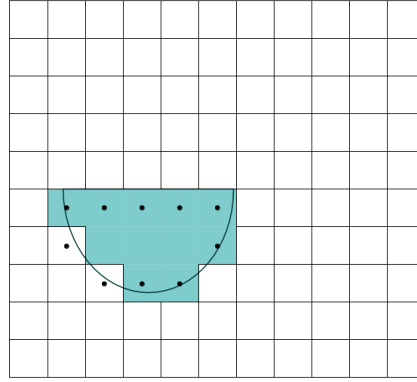
At the beginning every element in the grid is labeled 1, and every time that the model creates an object its respective grid cell should be relabeled. In order to do this the model must decide if the center of each cube is inside a channel or a crevasse.

Consider the center of the cube, say  $(x_0, y_0, z_0)$ . To facilitate the computations, the channel will be rotated back to its original position, that is, it will be rotated by  $\alpha$  in the clockwise sense, which is the channel's rotation. Also the channel will be translated so that its center will be at the origin. Therefore, the point  $(x_0, y_0, z_0)$  will be rotated and translated in the same way. Denote this new point by  $(x_0^*, y_0^*, z_0^*)$ .

Every cube whose third coordinate is above the third coordinate of the channel  $z_c$ , or below  $z_c - t$ , where  $t$  is the thickness, will not belong to the channel so their labels will not change. For the rest of the cubes the procedure will be as follows.



3.14(a): Plan view.



3.14(b): Cross sectional view at the fourth column.

Figure 3.14: Channel in-out function

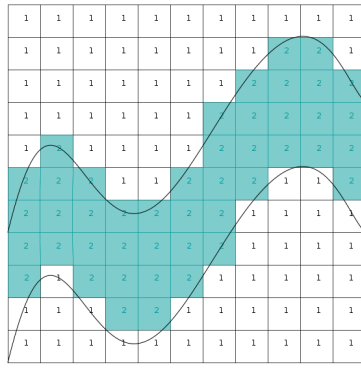
A function named **Channel in-out** will decide if the point  $(x_0, y_0, z_0)$  belongs to the channel (see Figure 3.14). Since the B-Spline curve defining the channel directrix is given by its functional form, then it is possible to find the second coordinate corresponding to  $x_0^*$ , that is  $y_c = f(x_0^*)$  where  $(x_0^*, f(x_0^*))$  belongs to the directrix.

Because the way the channel is constructed,  $x_0^*$  has a half ellipse associated to it. Then, to decide if the point  $(x_0^*, y_0^*, z_0^*)$  is inside that half ellipse the channel in-out function verifies if the next condition is valid:

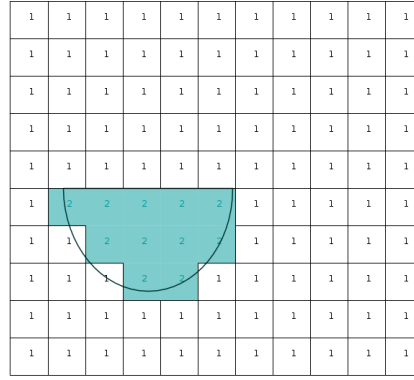
$$\frac{(y_c - y_0^*)^2}{\left(\frac{\omega}{2}\right)^2} + \frac{(z_c - z_0^*)^2}{t^2} - 1 > 0. \quad (3.4)$$

If the condition is false, then the center of the cube is in the channel so its corresponding entrance in the grid matrix will be multiplied by 2 as shown in Figure 3.15. If the condition is true, the model will analyse the crevasses associated to that channel. First each crevasse is rotated and translated so the crevasse's center will be at the origin, and the segment determining the crevasse's length will be over the  $x$ -axis. Again, the point is translated in the same way. Denote the new position by  $(x_0^{**}, y_0^{**}, z_0^{**})$ . Then the **Crevasse in-out** function determines if that point is inside the crevasse.

The crevasse in-out function works in a similar way as the channel in-out function. If  $x_0^{**} < 0$ ,  $x_0^{**} > l$ ,  $z_0^{**} > 0$  or  $z_0^{**} < -t$ , where  $l$  and  $t$  represent the crevasse's length and thickness respectively, then the point will not belong to



3.15(a): Plan view.



3.15(b): Cross sectional view at the fourth column.

Figure 3.15: Channel grid 3-D matrix.

the crevasse. So its entrance will remain the same.

Suppose now that none of those conditions are satisfied. If  $y_0^{**} > 0$ , then the auxiliary function will use the left side B-Spline curve to calculate the value  $y_c$  that corresponds to the  $x_0^{**}$  value on that curve. If  $y_0^{**} \leq 0$ , the same is done with the right side B-Spline curve. Since the crevasse's depth is also determined by a B-Spline curve in functional form then, it is possible to find the value  $z_t$  so  $(x_0^{**}, z_t)$  belong to the curve. Then, as before,  $x_0^{**}$  has half an ellipse associated to it. To decide if the point  $(x_0^{**}, y_0^{**}, z_0^{**})$  is inside that half an ellipse the crevasse in-out function verifies if the next condition is valid:

$$\frac{(y_c - y_0^{**})^2}{(y_c)^2} + \frac{(z_c - z_0^{**})^2}{z_t^2} - 1 > 0. \quad (3.5)$$

If the condition is false, then the center of the cube is in the crevasse so its entrance in the grid matrix will be multiplied by 3. If not, then the entrance will not be changed.

It may be necessary to erase an object to honor the constraints. In order to do this the labels generated by it must be removed. So the model will use the in-out functions mentioned before but instead of multiply by 2 or 3 it will multiply by 1/2 or 1/3.

### 3.2.3 Simulation Process

The first problem addressed by the model is to honor the well data. In order to do this it will construct channels, using the matrix described earlier, in the following way. Consider the first row of the matrix. The model will generate a channel using the first column and a perturbation of the second third columns as the channel center. The perturbation of the coordinate  $y$  is made by randomly choosing a number  $\beta$  between 0.75 and  $-0.75$ , multiplying it by  $\omega$  and adding it to  $y$ . For the perturbation of the coordinate  $z$  the model calculates the channel's thickness  $t'$  at the point  $(x, y, z_i)$  and add it to  $z_i$ , so the channel's center will be  $(x, y + \beta\omega, z_i + t')$ . If  $z^* = z_i + t' \leq z_f$ , then the value in the third column will be changed to  $z^{**} = z_i + t'$ . The process is repeated until the condition is no longer true.

When  $z^* > z_f$ , the model calculates two  $y$  coordinates over the ellipse where the channels thickness is  $h = z_2 - z^{**}$ , and determines the points  $(x, y_1, z^*)$  and  $(x, y_2, z^*)$  (see Figure 3.16). Finally the channel is translated so one of those points will be placed over the point  $(x, y, z_f)$  (see Figure 3.17), and the well data is honored. To indicate that this subinterval is already filled by channels the model changes the last column by 0, and continues to the next row until the last column of all rows are filled by zeros.

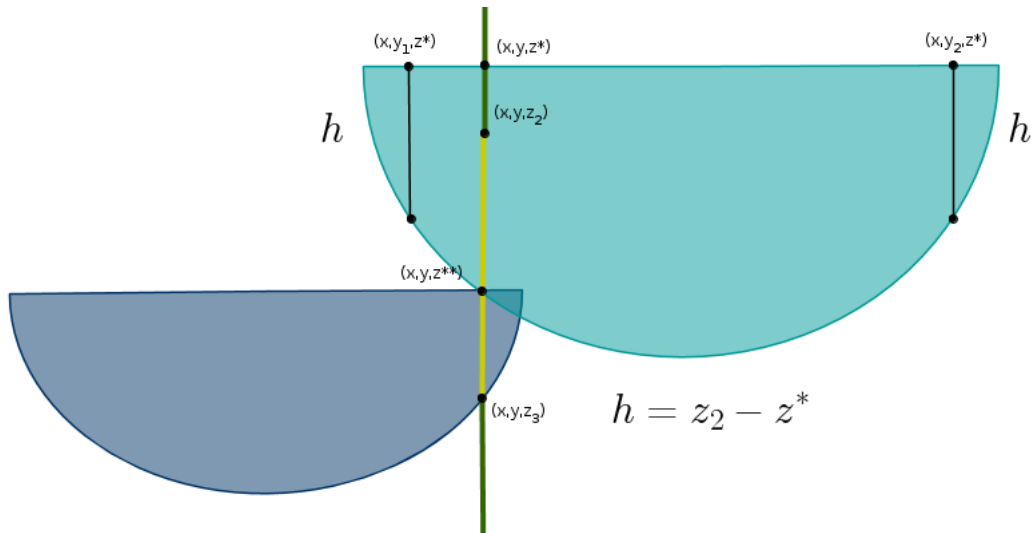


Figure 3.16: The condition is no longer true, so the model calculates the best  $y$ .

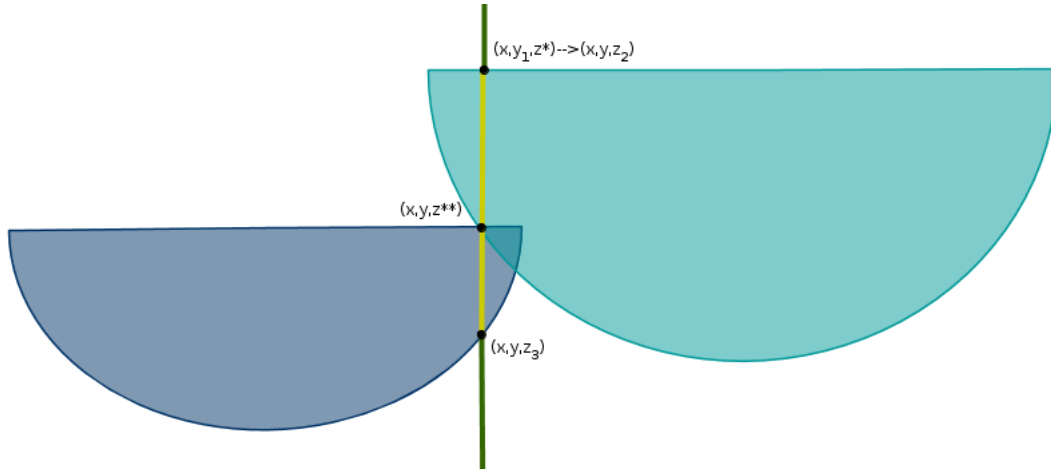


Figure 3.17: The channel is translated to honor the well data.

Before labeling the channel in the grid, it is essential to verify if it does not intersect any other well on the green intervals. To do this, the model will pick a well, and analyse the points  $(x_0, y_0, z_i)$  where  $x_0$ , and  $y_0$  are the first two coordinates of the well, and  $z_i$  is the  $z$  coordinate of the well, in the green intervals, determined by the grid partition of the third coordinate. If in the analysis of all the wells one of those points belongs to the channel or its crevasses, then this channel is rejected and the model will create a new one until one honors the well data. After this, the approved channel will be label on the grid.

Once the well data is honored, the model focuses in the problem of satisfying the global proportions. This problem is addressed by employing a simulated annealing algorithm.

The implementation of a simulated annealing algorithm requires an objective function. The objective function defines the matching criterion and measures the mismatch between the model and the observed data. It is an encoding of the constraints to be imposed on the model.

The global proportion will be denoted by  $P_g^*$  and will be given by  $P_g^* = v_d/v_t$  where  $v_d$  means the desired volume and  $v_t$  is the total volume of the area of interest. The objective function is given by

$$O = [P_g - P_g^*]^2, \quad (3.6)$$

where  $P_g$  represents the proportion obtained by the model. In other words if  $v$  is the volume calculated by the model, then

$$O(v) = \left( \frac{v - v_d}{v_t} \right)^2. \quad (3.7)$$

**Algorithm 2** Simulated Annealing Algorithm

---

```

1: procedure OPTIMIZATION
2:   SET { initial volume  $v$  };
3:   SET { initial value of objective function  $O(v)$  };
4:   SET { initial temperature  $t = 0.8$ };
5:   SET { annealing schedule  $\alpha = 0.99$ };
6:   while TRUE do
7:      $O_{old} = O(v)$ ;
8:     randomly choose an operation: 0= Erase, 1= Add;
9:     Add or Erase;
10:    SET { new volume  $v$  };
11:    test objective function,  $O_{new} = O(v)$ ;
12:    if  $O_{old} - O_{new} > 0$ , then
13:      continue;
14:    end if
15:    SET  $k = 5$ ,  $p = 9^k \cdot \log 2$ ,  $r = \exp p \cdot \left( \frac{O_{old} - O_{new}}{t} \right)$ ;
16:    if  $r \leq 0.5$ , then
17:      reject and undo;
18:    else accept;
19:    end if
20:    SET {after five operations new temperature  $t = \alpha t$ };
21:    UNTIL {20 consecutive attempts without acceptance };
22:  end while
23: end procedure

```

---

The initial value of the volume will be the volume of the channels already constructed for honoring the well data. Once this volume is calculated the objective function corresponding to that value is obtained.

There are two operations implemented by the model. One is **Add**, which refers to constructing an additional channel. If it is accepted, then a new channel will be labeled in the grid. The other one is **Erase**. This operation choose one channel to erase, between all the channels added after the well data is honored. That is, none of the channels added to honor the well data will be affected by the optimization process<sup>5</sup>. If this operation is approved, that channel will be unmarked from the grid.

The model will randomly choose an operation and analyse if the objective function is lower. If it is, the channels operation is approved and the labels on the grid will change depending on what operation was accepted. If

---

<sup>5</sup>This can be done because it is assumed that the global proportion is consistent with the well data, i.e. the volume desired will be bigger than the one required to fill the wells.

not, the method decides if the objective function is low enough by analysing the value of  $r$ . If it is low enough, the operation is accepted. Otherwise, the process starts again.

After five operations are accepted, the temperature is updated by the annealing schedule, and the process continues until the objective function is lower than the error. If no operation is accepted after 20 tries, then the algorithm stops.

Because of how the method works, it is possible to do only one of the processes. This means that the program could only honor the well data, assuming that there is no global proportion so the optimization method is not applied. Or it could only honor the global proportions, assuming there is no well data, and that the initial volume for the optimization method will be zero.

The simulation process is summarize in the flux chart presented in 3.18, while 3.19 shows the process implemented to honor the well data. Remember that the simulation process is presented in 2.5.



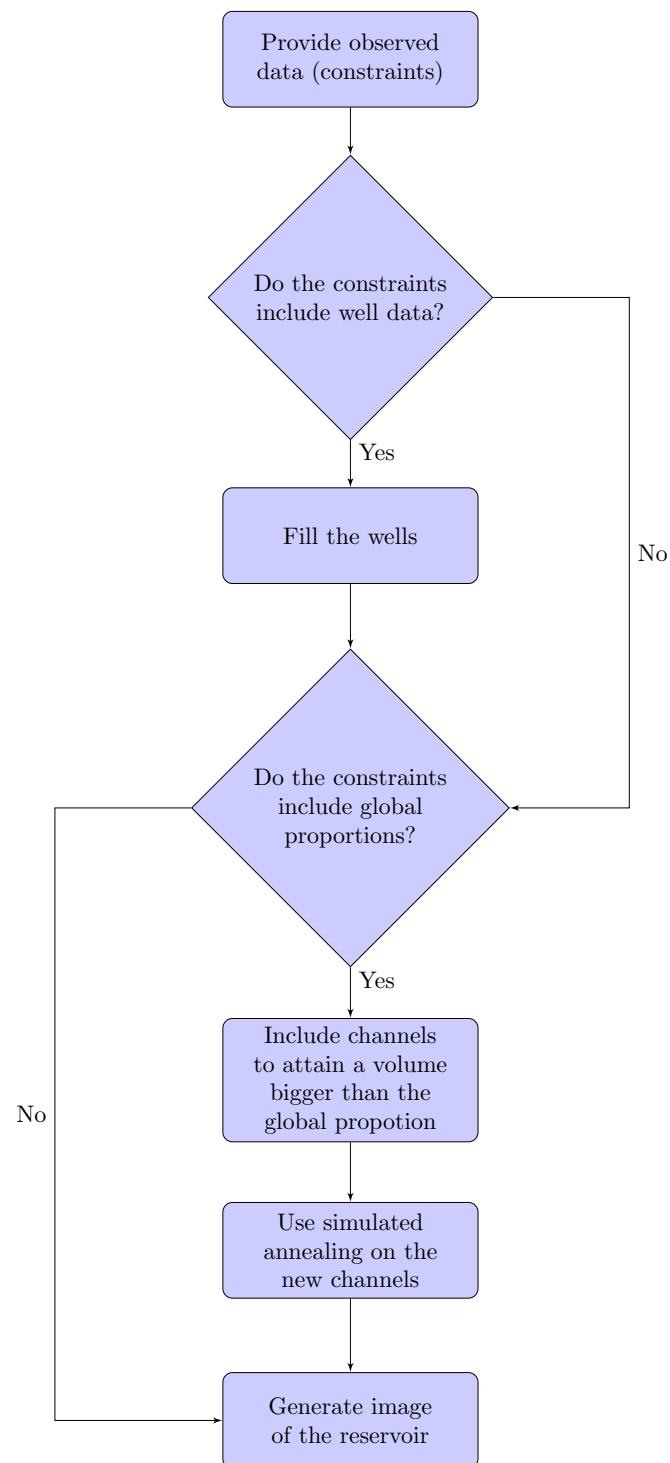


Figure 3.18: Modeling process

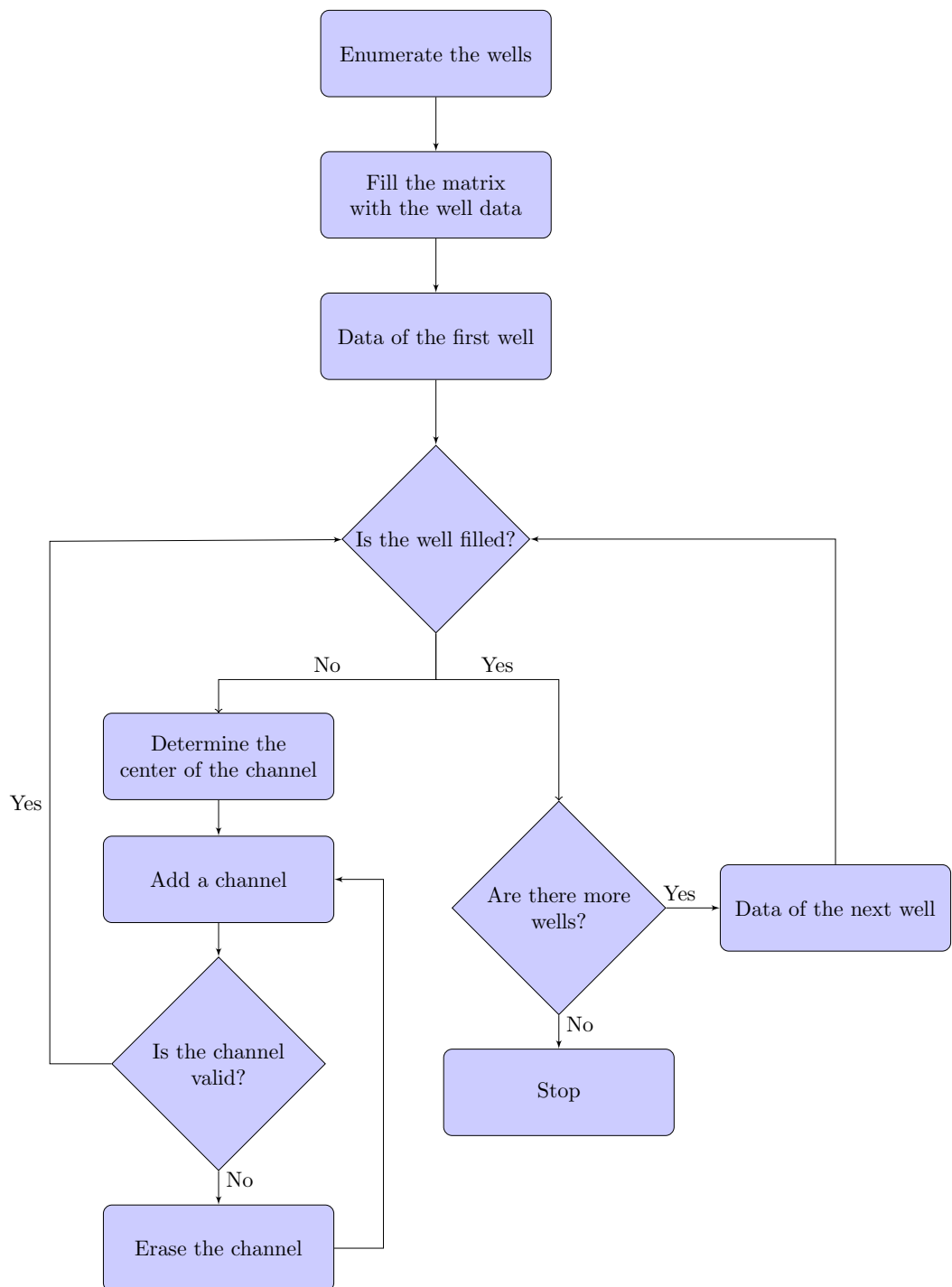


Figure 3.19: Filling wells process

## 4

## Implementation Results

This chapter presents some results obtained by the model presented in the previous chapter. The first section covers the construction of the objects, while the second one shows the results obtained after different simulations.

### 4.1

#### Objects Construction

The simulation domain is  $[x_{min}, x_{max}] \times [y_{min}, y_{max}] \times [z_{min}, z_{max}]$ . In particular, we used the cartesian product of the interval  $[-5, 5]$ . The simulation mesh grid employed here is  $100 \times 100 \times 100$ .

Unless stated otherwise, the width and thickness of the channel will be randomly chosen in the intervals  $[0.2, 1.2]$  and  $[0.2, 0.7]$  respectively. The crevasse's width and length will also be chosen from the interval  $[0.2, 1.2]$ . The angles for the channel and crevasses will be random numbers in the intervals  $[\pi/6, \pi/3]$  and  $[0, \pi/4]$  respectively.

For a better understanding of the construction process we will present the B-Splines defining every object, as well as the image after applying the in-out functions.

#### 4.1.1

##### Crevasse Object

In the previous chapter, it was mentioned that the construction of the crevasses depends on three B-Splines: two defining the boundaries and one defining the depth of the crevasse. Those B-Splines are presented in Figure 4.1. When the B-Splines are constructed, the Crevasse in-out function determines which cubes belong to the crevasse and marks them in the mesh. Figure 4.2

shows how the algorithm displays this image.

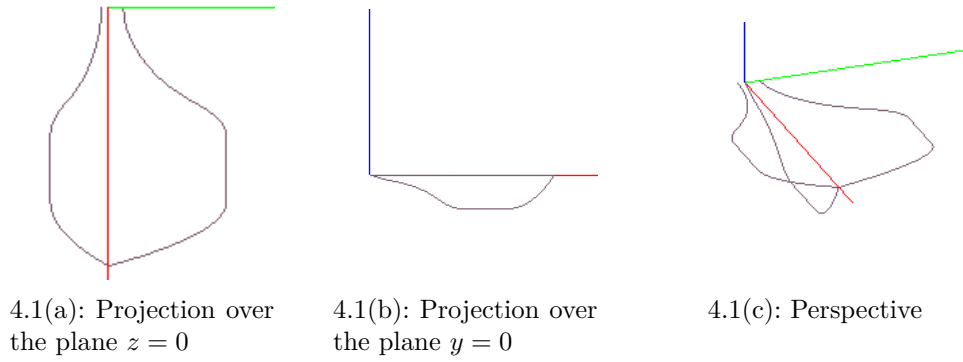


Figure 4.1: Construction of the crevasse

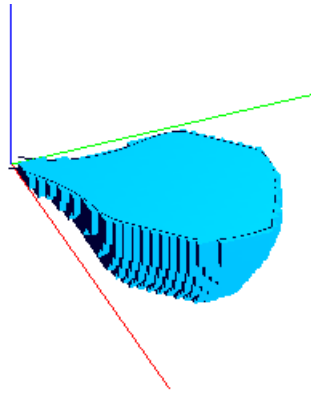


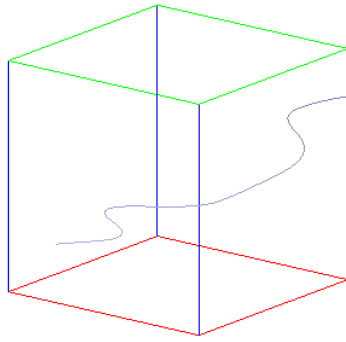
Figure 4.2: Crevasse in-out function

#### 4.1.2 Channel Object

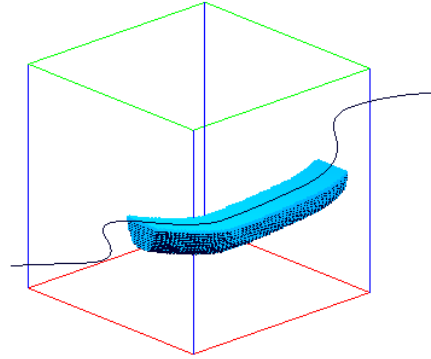
The B-Spline defining each channel is defined over a directrix line long enough so they pass through the cube. This causes that some parts of the channel will be located outside the area of interest, but this ensures that the channel does not end in the middle of the domain (see Figure 4.3).

Figure 4.4 shows the construction of both objects. The program chooses, given a distribution, how many crevasses are associated to each channel. However, since part of the channel can be located outside the area of interest,

it is possible that some of those crevasses remain outside the cube as well.

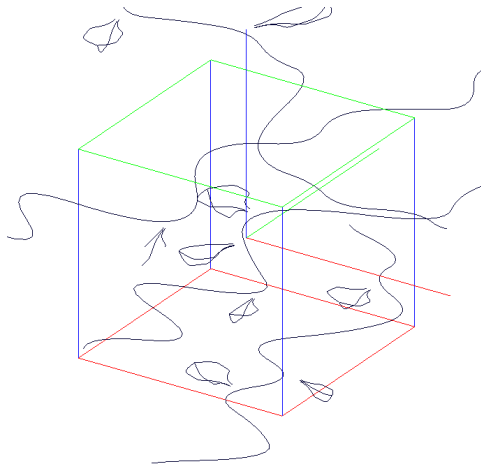


4.3(a): B-Splines construction

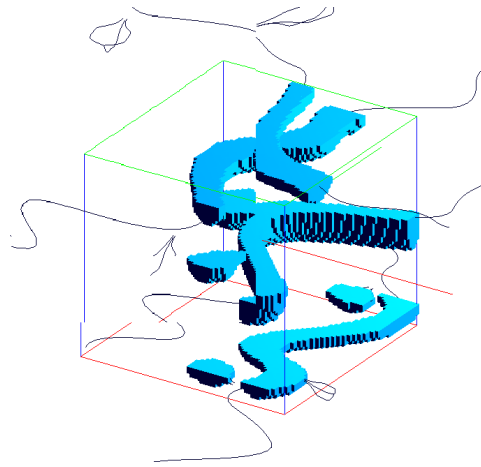


4.3(b): Channel in-out function

Figure 4.3: Construction of a channel



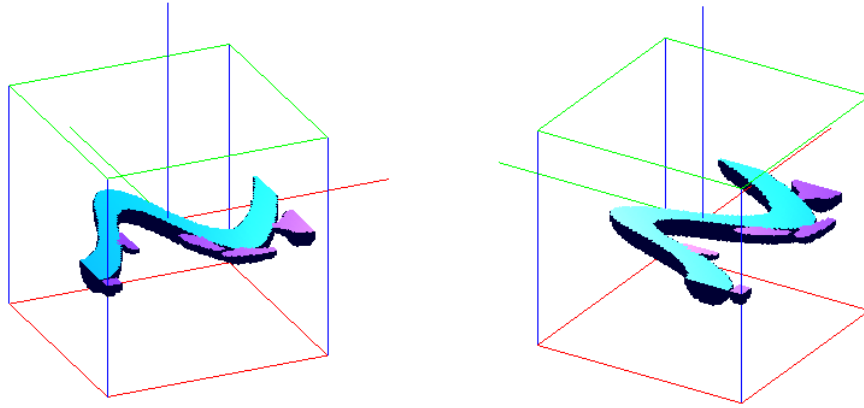
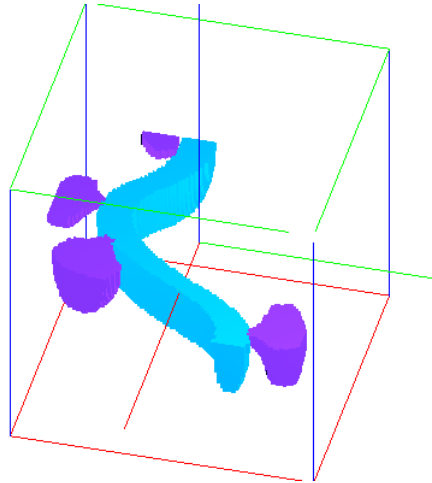
4.4(a): B-Splines construction



4.4(b): Channel and Crevasse in-out functions

Figure 4.4: Construction of a channel with its respective crevasses

The program allows the construction of just one channel at a time including its crevasses by defining its width and thickness. The next figures are examples of this. Figure 4.5 displays a channel with width 2 and thickness 0.75, and Figure 4.6 shows a channel with width 1 and thickness 1.8.

Figure 4.5: Channel with width  $\omega = 2$  and thickness  $t = 0.75$ Figure 4.6: Channel with width  $\omega = 1$  and thickness  $t = 1.8$ 

## 4.2 Simulations

The program has two constraints and it is possible to make a simulation including only one of them or both. The following sections exhibit those three types of simulations.

The *Well Data* section shows one simulation including only well constraints. The *Global Proportion* section presents simulations conditioning only the volume of the reservoir. Finally, the last section covers the simulations including both constraints.

### 4.2.1 Well Data

Figure 4.7 presents how the program locates the channels to honor the well data. As mention before, the program will only accept a channel if it respects all the wells. This means that the channel only hits the wells in the yellow intervals.

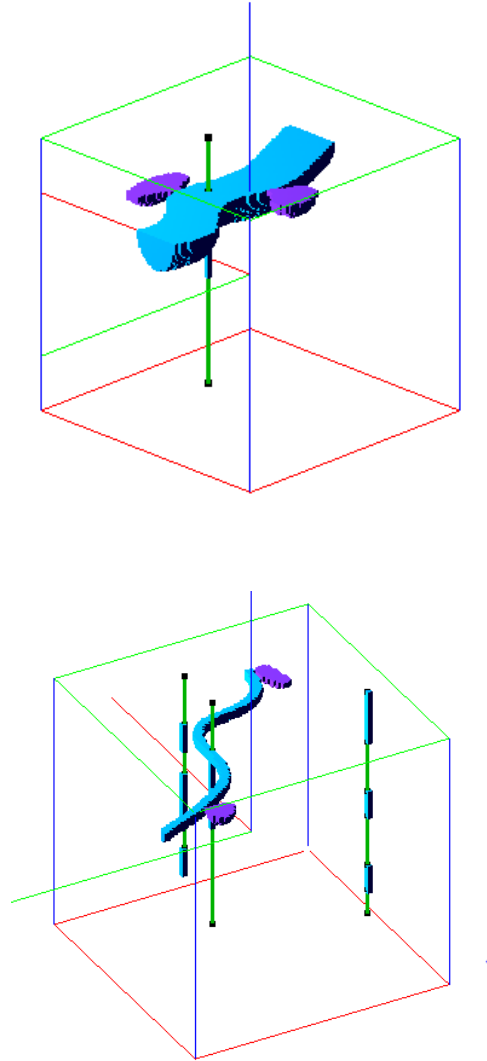


Figure 4.7: Channel honoring well data

The first simulation presents a reservoir conditioned to one well. That well presented tree yellow intervals which the program fills completely. This simulation is presented in Figure 4.8.

The second simulation, presented in Figure 4.9, had three wells. Two of the wells with three green intervals and three yellow ones, while the third well had two green intervals and only a yellow one.

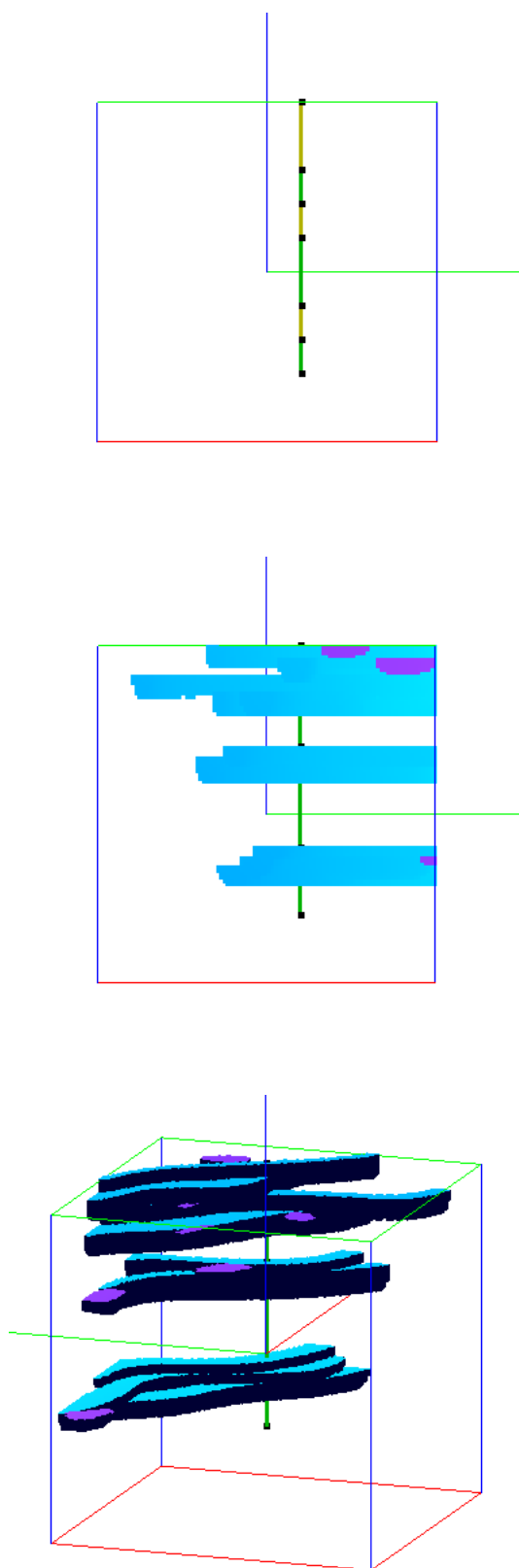


Figure 4.8: Simulation with one well.



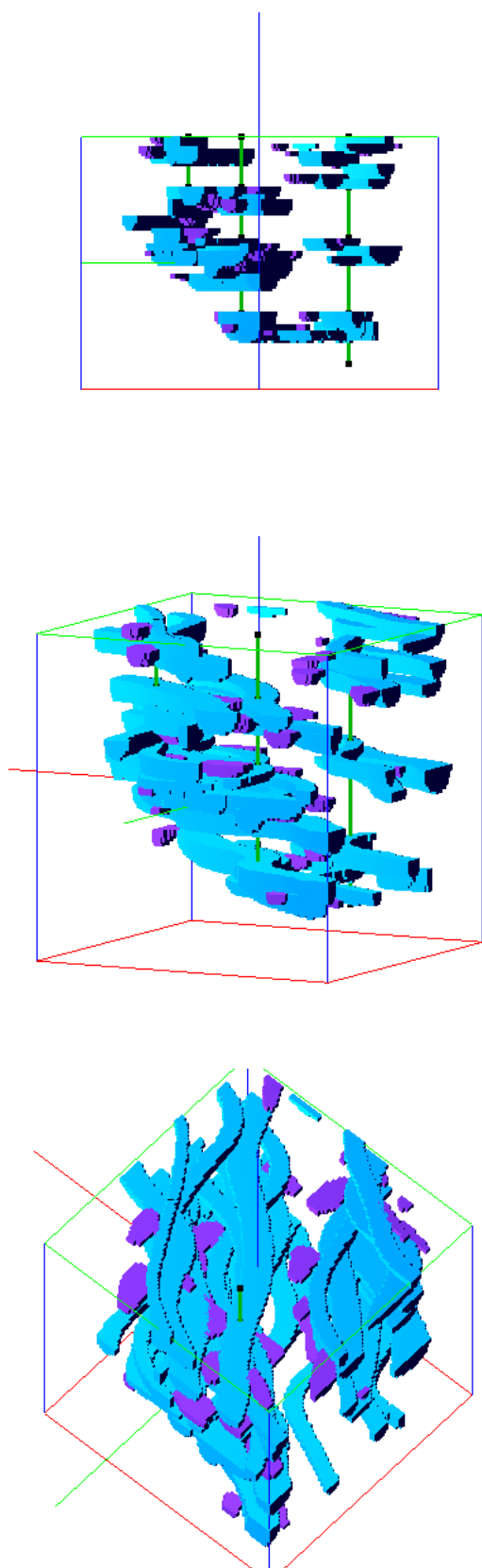


Figure 4.9: Simulation with three wells

The program also allows the user to determine the number of crevasses associated to each channel. In the next simulation we set the program to randomly choose the number of crevasses in the set  $\{7, 8, 9, 10, 11\}$ . This time the simulation also had three wells and it is presented in Figure 4.10.

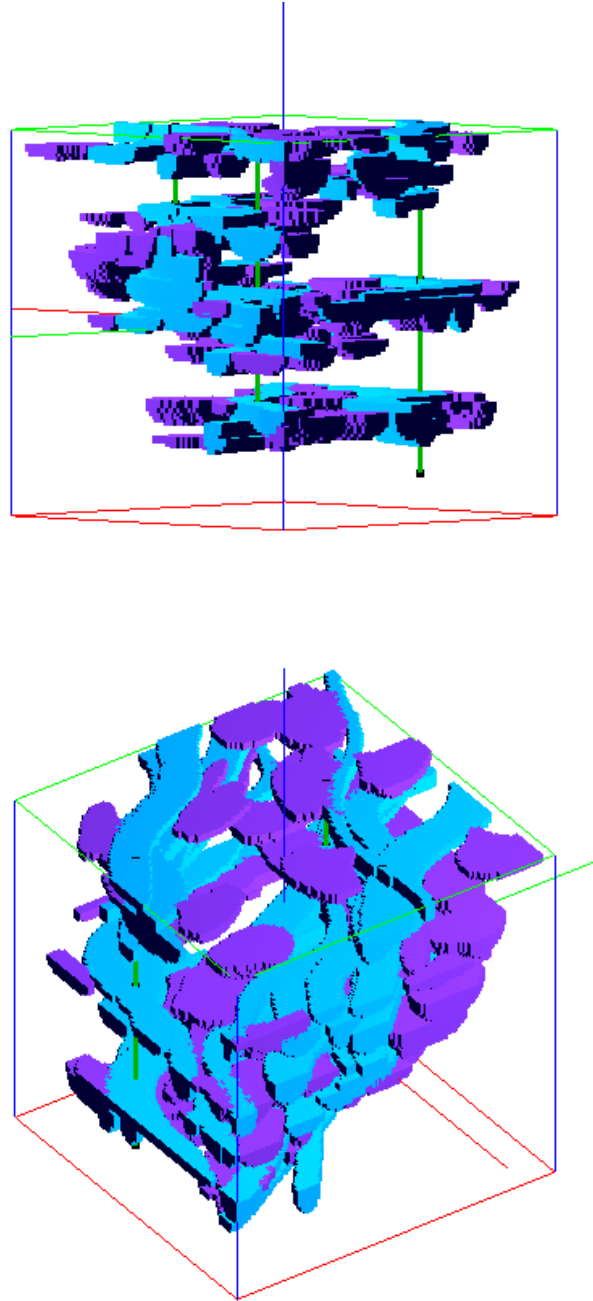


Figure 4.10: Three wells and more crevasses

### 4.2.2

#### Global Proportions

The simulations exhibited in this section only condition the volume of the reservoir. The first simulation is shown in Figure 4.11. Its global proportion is 10% of the total volume. That means that the program marked 10% of the cubes in the grid.

For this simulation the program employed the Simulated Annealing algorithm. The program process is shown in Table 4.1 and Table 4.2 shows the volume of some selected sections as well as their average volume.

Annealing schedule	$\alpha = 0.9$
Initial channels	51
Volume before SA	10.1901%
Final $f(t)$	0.00862021
Final $O(v)$	$4.7524 \times 10^{-8}$
Final channels	52
Final volume	9.9782%
Duration SA	12'28"
Total Duration	14'16"

Table 4.1: Simulation process

Section	Volume
0	0.0487
8	0.0911
16	0.101
24	0.0977
32	0.0615
40	0.1191
48	0.1029
56	0.1077
64	0.0924
72	0.1498
80	0.1274
88	0.1231
96	0.0982
Average	0.1016

Table 4.2: Sectional average volume

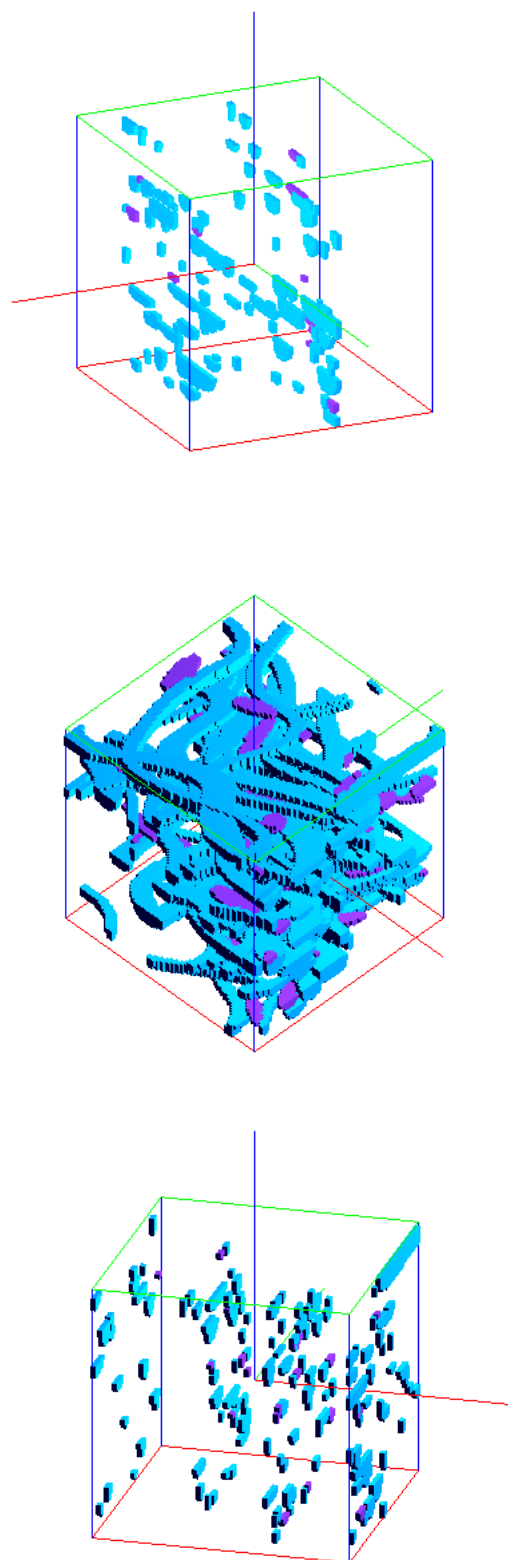


Figure 4.11: Global proportion of 10%

The simulation presented in Figure 4.12 is conditioned to a global proportion of 30%. The simulation processes shown in Table 4.3, and Table 4.4 presents the sectional average volume.

Annealing schedule	$\alpha = 0.9$
Initial channels	170
Volume before SA	30.1133%
Final $f(t)$	0.00219114
Final $O(v)$	$2.5921 \times 10^{-8}$
Final channels	178
Final volume	29.9839%
Duration SA	19'49"
Total Duration	27'15"

Table 4.3: Simulation process

Section	Volume
0	0.2489
8	0.275
16	0.2615
24	0.2957
32	0.3147
40	0.3361
48	0.3394
56	0.3125
64	0.3359
72	0.3131
80	0.3066
88	0.3087
96	0.273
Average	0.3016

Table 4.4: Sectional Average

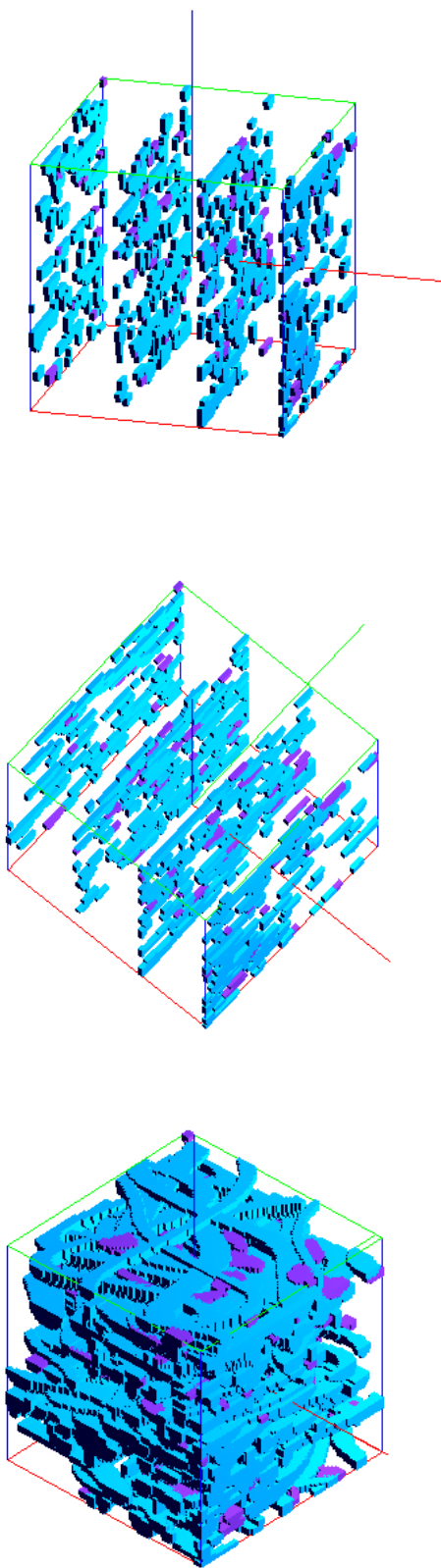


Figure 4.12: Global proportion of 30%

### 4.2.3

#### Both Constraints

The simulation shown in Figure 4.13 has both constraints. It has two wells each one with three green intervals and three yellow ones. The global proportion is 6.5%.

At first the program will honor the well data, then it has to continue adding channels until it passes the objective volume. This process is detailed in Table 4.5, while Table 4.6 shows the sectional average volume.

Annealing schedule	$\alpha = 0.9$
Channels conditioned to wells	26
Initial channels	39
Volume before SA	6.8931%
Final $f(t)$	0.0305216
Final $O(v)$	$4.61041 \times 10^{-7}$
Final channels	32
Final volume	6.5679%
Duration SA	9'52"
Total Duration	11'23"

Table 4.5: Simulation process

Section	Volume
0	0.0446
8	0.0646
16	0.0635
24	0.0634
32	0.0824
40	0.0997
48	0.0562
56	0.0454
64	0.0514
72	0.0644
80	0.0969
88	0.0701
96	0.0476
Average	0.0654

Table 4.6: Sectional Average

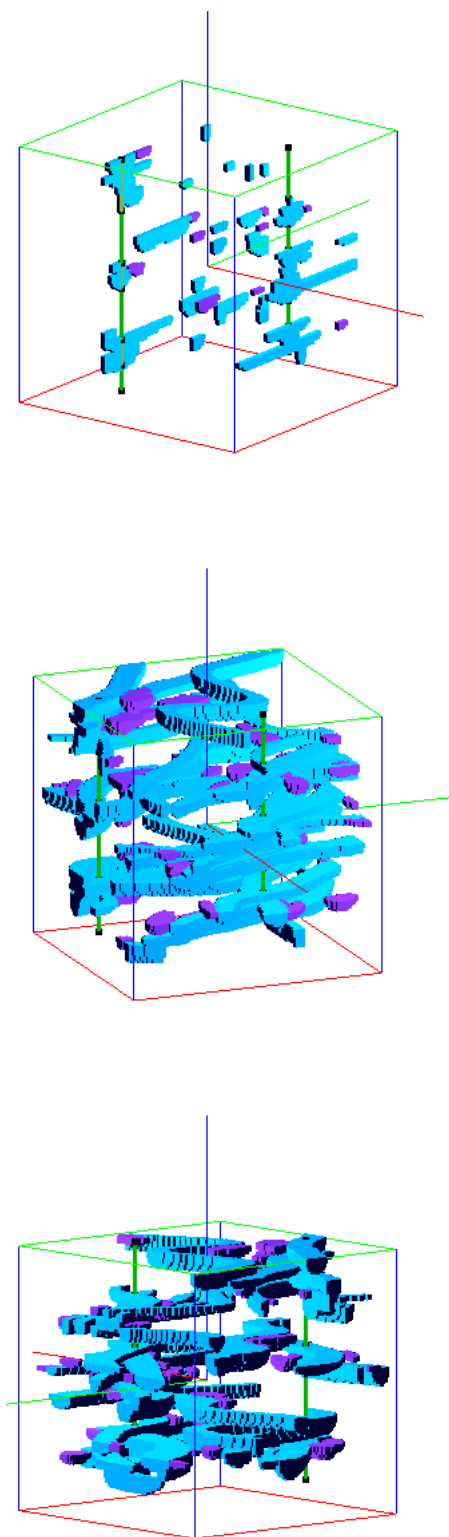


Figure 4.13: Simulation with two wells and 6.5% global proportion



## 5

### Conclusion

In this work we discussed the development of a stochastic modeling technique involving three-dimensional geological objects. Those objects represent two of the main facies present in a fluvial reservoir, namely the channels and its crevasses. Those objects were constructed based on nonparametric B-Spline curves presented in chapter 2, making use of a regular grid and a half ellipse geometric form for the cross section of the objects. This procedure was explained in chapter 3.

The parametrization used to model the channel is based on the procedure implemented in PetBool [Poletto, 1996] and FLUVSIM [Deutsch and Tran, 2002]. However, the parametrization of the crevasse is not based on the FLUVSIM model. The approach that we use is based on the channel parametrization, with the difference that for the crevasse we allow the objects to be non symmetrical. This is accomplished by using two different B-Spline curves to construct its boundary, which are determined by two different widths. This is the main contribution of this work.

The model allows the user to manipulate the parameters that determine each object, as well as the number of crevasses corresponding for each channel. This enables the user to create narrower or wider channels with few or a lot of crevasses depending on the experimental histograms available.

An optimization method using simulated annealing technique was discussed and utilized to ensure that the model simulated reflects the observed data as closely as possible. In chapter 4 we presented the successful implementation of the algorithms developed in this work in different scenarios, and how the program can be manipulated to satisfy the constraints.

In future works we pretend to use parallel simulation to improve the CPU

cost. As proposed by Deutsch and Tran [Deutsch and Tran, 2002], another research area includes the simulation of more facies like the levees. Also, more constraints can be included in the simulation process. For example, the use of vertical proportion curves to specify the proportion of every facies type. The importance of the implementation of the simulated annealing algorithm becomes more clear with the inclusion of more constraints. This is because more constraints implies a more complicated objective function.

## Bibliography

- [Allen, 1978] Allen, J. (1978). Studies in fluvial sedimentation: An exploratory quantitative model for the architecture of avulsion-controlled alluvial suites. *Sedimentary Geology*, 21(2):129–147.
- [Barreto et al., 1995] Barreto, A., Farias, R., Krueel, R., Lanzarini, W., Lopes, H., Pesco, S., and Tavares, G. (1995). Petbool: simulador estocástico da geometria e heterogeneidades de reservatórios petrolíferos usando o método booleano. Primeiro Workshop em Simulação e Engenharia de Reservatórios.
- [Deutsch and Tran, 2002] Deutsch, C. and Tran, T. (2002). Fluvsim: a program for object-based stochastic modeling of fluvial depositional systems. *Computers & Geosciences*, 28(4):525–535.
- [Deutsch and Wang, 1996] Deutsch, C. V. and Wang, L. (1996). Hierarchical object-based stochastic modeling of fluvial reservoirs. *Mathematical Geology*, 28(7):857–880.
- [Deviese, 2010] Deviese, E. (2010). *Modeling Fluvial Reservoir Architecture using Fluvial Process*. PhD thesis, TU Delft, Delft University of Technology.
- [Ezekiel et al., 2013] Ezekiel, E., Abowei, J., and Charles, E. (2013). Effects of flooding on amassoma flood plain sediments niger delta, nigeria. *App. Sci. Report*, 4(1):173–180.
- [Farin, 2002] Farin, G. (2002). *Curves and Surfaces for CAGD: A Practical Guide*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 5th edition.
- [Farin, 1996] Farin, G. E. (1996). *Curves and Surfaces for Computer-Aided Geometric Design: A Practical Code*. Academic Press, Inc., Orlando, FL, USA, 4th edition.

- [Haldorsen and Lake, 1982] Haldorsen, H. and Lake, L. (1982). A new approach to shale management in field scale simulation models. Technical report, Univ. of Texas.
- [Hall, 2007] Hall, C. (2007). *Introduction to the Geology of Southern California and Its Native Plants*. University of California Press.
- [Helbig, 2015] Helbig, K. (2015). *Modeling The Earth For Oil Exploration: Final Report of the CEC's Geoscience I*. Elsevier Science.
- [Hopgood, 2001] Hopgood, A. A., editor (2001). *Intelligent Systems for Engineers and Scientists (2Nd Ed.)*. CRC Press, Inc., Boca Raton, FL, USA.
- [Johnson et al., 1984] Johnson, H., Krol, D., et al. (1984). Geological modeling of a heterogeneous sandstone reservoir: Lower jurassic statfjord formation, brent field. In *SPE Annual Technical Conference and Exhibition*. Society of Petroleum Engineers.
- [Kirkpatrick et al., 1983] Kirkpatrick, S., Gelatt, C. D., Vecchi, M. P., et al. (1983). Optimization by simulated annealing. *science*, 220(4598):671–680.
- [Longwell, 1949] Longwell, C. R. (1949). *Sedimentary facies in geologic history: conference at meeting of the Geological Society of America held in New York, New York, November 11, 1948*, volume 39. Geological Society of America.
- [Ma et al., 2011] Ma, Y. Z., La Pointe, P. R., et al. (2011). *Uncertainty Analysis and Reservoir Modeling: Developing and Managing Assets in an Uncertain World, AAPG Memoir 96*, volume 96. AAPG.
- [Menezes, 2004] Menezes, L. (2004). Mapeamento digital de análogos a reservatórios petrolíferos: exemplos para depósitos fluviais da unidade açu-3-bacia potiguar. Master's thesis, Universidade Federal do Rio Grande do Norte.
- [Miall, 2006] Miall, A. (2006). *The Geology of Fluvial Deposits: Sedimentary Facies, Basin Analysis, and Petroleum Geology*. Springer Berlin Heidelberg, 4th edition.
- [Moore, 1949] Moore, R. C. (1949). Meaning of facies. *Geological Society of America Memoirs*, 39:1–34.
- [Neuendorf, 2005] Neuendorf, K. K. E. (2005). *Glossary of Geology*. American Geological Institute.

- [Phan, 2002] Phan, V. (2002). *Modeling Technique for Generating Fluvial Reservoir Descriptions Conditioned to Static and Dynamic Data*. PhD thesis, Stanford University.
- [Piegl and Tiller, 1997] Piegl, L. and Tiller, W. (1997). *The NURBS Book (2Nd Ed.)*. Springer-Verlag New York, Inc., New York, NY, USA.
- [Poletto, 1996] Poletto, C. A. (1996). Modelagem geológica orientada a objetos e simulação estocástica da geometria de reservatórios fluviais. Master's thesis, Universidade Estadual de Campinas.
- [Pyrcz and Deutsch, 2014] Pyrcz, M. J. and Deutsch, C. V. (2014). *Geostatistical Reservoir Modeling*. Oxford University Press, second edition edition.
- [Sánchez-Reyes, 1994] Sánchez-Reyes, J. (1994). Single-valued tubular patches. *Computer Aided Geometric Design*, 11(5):565–592.
- [Tavares et al., 2002] Tavares, G., Lopez, H., Pesco, S., and Poletto, C. A. (2002). Petbool: a software for stochastic modeling of geological objects. In *Proceedings of the 8th IAMG Annual Conference, Berlin*, volume 1, pages 203–208.
- [Tvelin, 2016] Tvelin, S. (2016). This old earth. [http://www.thisoldearth.net/Geology\\_Online-1\\_Subchapters.cfm?Chapter=5&Row=4](http://www.thisoldearth.net/Geology_Online-1_Subchapters.cfm?Chapter=5&Row=4). Accessed: 10/02/2016.