PROGRAMA DE PÓS-GRADUAÇÃO EM INFORMÁTICA

Universidade Federal do Rio de Janeiro

WAGNER DOS SANTOS VIEIRA

A SEMI-AUTOMATIC FRAMEWORK TO IDENTIFY BRIGHT SPOTS AS POTENTIAL HYDROCARBON INDICATORS IN SEISMIC DATA, USING AN MPPDB INFRASTRUCTURE AND ROUGH SET THEORY

RIO DE JANEIRO
2017

Instituto de Matemática

Instituto Tércio Pacitti de Aplicações e Pesquisas Computacionais

UNIVERSIDADE FEDERAL DO RIO DE JANEIRO
INSTITUTO DE MATEMÁTICA
INSTITUTO TÉRCIO PACITTI DE APLICAÇÕES E PESQUISAS COMPUTACIONAIS
PROGRAMA DE PÓS-GRADUAÇÃO EM INFORMÁTICA

WAGNER DOS SANTOS VIEIRA

A Semi-Automatic Framework to Identify Bright Spots as Potential Hydrocarbon Indicators in Seismic Data, using an MPPDB Infrastructure and Rough Set Theory

A dissertation submitted in partial fulfillment of the requirements for the degree of Master (Computer Science, Information Systems) in Programa de Pós-Graduação em Informática, Universidade Federal do Rio de Janeiro.

Advisors: Paulo de Figueiredo Pires
Fábio André Perosi

Rio de Janeiro
2017

FICHA CATALOGRÁFICA

Vieira, Wagner dos Santos
V657s A Semi-Automatic Framework to Identify Bright Spots as Potential Hydrocarbon Indicators in Seismic Data, using an MPPDB Infrastructure and Rough Set Theory / Wagner dos Santos Vieira. -- Rio de Janeiro, 2017.
74 f.

Orientador: Paulo de Figueiredo Pires.
Coorientador: Fábio André Perosi.
Dissertação (mestrado) - Universidade Federal do Rio de Janeiro, Instituto Tércio Pacitti de Aplicações e Pesquisas Computacionais, Programa de Pós-Graduação em informática, 2017.

1. Análise de dados. 2. Rough Set. 3. Processo de Suporte. 4. Petróleo & gás. 5. Processamento paralelo. I. Pires, Paulo de Figueiredo, orient. II. Perosi, Fábio André, coorient. III. Título.

WAGNER DOS SANTOS VIEIRA

A Semi-Automatic Framework to Identify Bright Spots as Potential Hydrocarbon Indicators in Seismic Data, using an MPPDB Infrastructure and Rough Set Theory

A dissertation submitted in partial fulfillment of the requirements for the degree of Master (Computer Science, Information Systems) in Programa de Pós-Graduação em Informática, Universidade Federal do Rio de Janeiro.

Prof. D.Sc. Paulo de Figueiredo Pires – Advisor
COPPE/UFRJ, Brasil

UFRJ/PPGI

Prof. D.Sc. Fábio André Perosi – co-Advisor
IAG/USP, Brasil

UFRJ/IGEO

Prof. D.Sc. Claudio Miceli de Farias
PPGI/UFRJ, Brasil

UFRJ/NCE

Prof. D.Sc. Fábio André Machado Porto
PUC Rio, Brasil

LNCC

Prof. D.Sc. Angelo Ernani Maia Ciarlini
PUC Rio, Brasil

Dell EMC

Rio de Janeiro

2017

ACKNOWLEDGEMENTS

Agradeço primeiramente a Deus, responsável pela minha vida e por prover tudo que precisei para seguir em frente com meus estudos.

Agradeço aos meus pais por apoiarem meus projetos e que por várias vezes se deslocaram para o estado do Rio de Janeiro, para me darem suporte nos momentos em que eu mais precisava.

Agradeço aos meus amigos no Rio de Janeiro, que foram uma família para mim, em especial ao meu grande amigo Diogo e ao Elói que me trouxe de volta à vida acadêmica.

Agradeço ao Dr. Jairo, que nos últimos sete meses tão bem cuidou de minha saúde, permitindo que eu pudesse seguir em frente com minha vida, estudos e trabalho.

Agradeço ao meu orientador, Prof. Paulo de Figueiredo Pires, por acreditar no meu trabalho, pela paciência que teve comigo, por me orientar durante todo o trabalho da dissertação e pelo conhecimento passado em sala de aula.

Agradeço ao meu coorientador, Fábio André Perosi, parceria de grande importância para meu mestrado, que tão bem complementou o conhecimento científico necessário para esse trabalho, provendo todo o suporte na areas de geofísica e geologia.

Agradeço aos meus colegas do UBICOMP pela ajuda prestativa durante a realização deste trabalho, em especial ao meu colega e amigo Bruno. Também agradeço ao Jonas Dias, Edward Pacheco (Dell EMC) e Brian de la Motte (Zdata) por esclarecerem dúvidas que tive neste trabalho.

Agradeço à Universidade Federal do Rio de Janeiro por me acolher como aluno, e aos docentes que tive a satisfação de conhecer no Programa de Pós-Graduação em Informática da UFRJ.

Agradeço à Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES) pela concessão de bolsa durante dois anos do período de realização deste mestrado.

This research was developed with financial resources of the partnership between the Federal University of Rio de Janeiro and CAPES (Brazilian Funding Agency).

“O saber a gente aprende com os mestres e os livros.
A sabedoria se aprende é com a vida e com os humildes”

(*Cora Coralina)*

“Ninguém ignora tudo. Ninguém sabe tudo.
Todos nós sabemos alguma coisa. Todos nós ignoramos alguma coisa.
Por isso aprendemos sempre"

(Paulo Freire)

RESUMO

Este trabalho descreve um arcabouço com um processo computacional para auxiliar intérpretes durante a atividade de interpretação sísmica, detectando *bright spots* como possíveis indicadores de hidrocarbonetos em dados sísmicos. Esse conjunto de dados compreende um conjunto de atributos sísmicos, uma vez que tais dados podem ser vastos, a interpretação sísmica manual pode ser difícil ou mesmo inviável, assim nosso arcabouço surge para tratar esse problema. Nossa proposta usa a teoria de *Rough Set* e uma infra-estrutura de processamento massivamente paralelo. Através de *Rough Set*, podemos tentar reduzir o conjunto de atributos sísmicos e criamos um conjunto de regras de classificação para detectar *bright spots*. Para gerar as regras de classificação, executamos um processo de treinamento supervisionado via *Rough Set* na implementação do nosso arcabouço, incorporando conhecimento de especialista, o que aumenta a acurácia de nossos resultados. Em relação ao desempenho, nosso arcabouço pode ser melhor do que outra proposta, uma vez que distribuímos adequadamente os dados sísmicos em um banco de dados de processamento massivamente paralelo. Outra vantagem que nosso arcabouço tem sobre outra abordagem é: nossas regras de classificação podem ser genéricas para diferentes conjuntos de dados sísmicos. Assim, a geração de regras ocorre apenas uma vez, que é na implementação do arcabouço. Uma vez que mantemos o mesmo conjunto dessas regras, para garantir flexibilidade ao nosso arcabouço de suporte à interpretação sísmica, fornecemos ao usuário um conjunto de parâmetros de entrada, que permite ajustar a consulta para detectar *bright spots*. Por fim, experimentos preliminares, realizados com dados sísmicos reais da costa holandesa, demonstram que nosso arcabouço é capaz de identificar *bright spots* relacionados à acumulação de hidrocarbonetos.

Palavras-chave: Análise de Dados. Processo de Suporte. Rough Set. Processamento Paralelo. Interpretação Sísmica. Petróleo & Gás.

ABSTRACT

This work describes a framework with a computational process to assist interpreters during the seismic interpretation activity, detecting bright spots as potential hydrocarbon indicators within a seismic dataset. Such dataset comprises a set of seismic attributes, and since the dataset can be vast the manual seismic interpretation can be hard or even unviable, thus our framework emerges to tackle this problem. Our proposal uses Rough Set theory and a massively parallel processing infrastructure. Through Rough Set, we can attempt to reduce the set of seismic attributes and, we create a set of classification rules to detect bright spots. To achieve the classification rules, we run a Rough Set's supervised training process in the framework implementation, incorporating specialist's knowledge, which increases the accuracy of our results. Regarding performance, our framework further may be faster than another proposal since we properly distributed the seismic data in a massively parallel processing database. Another advantage that our framework has over another approach is: our classification rules may be generic for any seismic dataset. Thus, the rules generation occurs just once at our framework implementation. Since we always keep the same set of rules, to guarantee flexibility to our seismic interpretation support framework, we provide to the interpreter user a set of input parameters, which allows tuning the pattern query for detect bright spots. Finally, preliminary experiments running on real seismic data of Dutch coast show that our framework can identify bright spots related to hydrocarbon accumulation.

Keywords: Data Analysis. Support Process. Rough Set. Parallel Processing. Seismic Interpretation. Oil & Gas.

LIST DE FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

2D – *Two Dimensions*

3D – *Three Dimensions*

ANN – *Artificial Neural Network*

API – *Application Programming Interface*

AVO – *Amplitude Versus Offset*

CSV – *Comma-separated values*

CV – *Coefficient of Deviation*

DDL – *Data Definition Language*

MPPDB – *Massively Parallel Processing Database*

MST – *Minimum-Spanning Tree*

PCA – *Principal Component Analysis*

SD – *Standard Deviation*

SEG – *Society of Exploration Geophysicists*

SEG-Y – *A seismic file standard type*

SQL – *Structured Query Language*

UML – *Unified Modeling Language*

TABLE OF CONTENTS

1 INTRODUCTION

In oil & gas exploration, the geological analysis is a process to verify the structures of a given area aiming at finding a potential accumulation of hydrocarbons. The seismic volume is one of the most used data sources to perform such geological analysis (FIGUEIREDO 2007). The data of a seismic volume originates from an acquisition method called seismic reflection, which scans the subsoil layers to produce a set of time series representing the seismic waves commonly called seismic traces. Once acquired, the seismic volume is processed to remove noises, which are interferences that change the amplitude values of the seismic traces. After the processing, it is still possible to have remaining noises, because some of them are quite close to the seismic traces then such noises can be mistaken with the seismic traces. Finally, the seismic volume is interpreted in the so-called seismic interpretation phase (THOMAS 2004), which aims at identifying seismic patterns related to hydrocarbons presence.

1.1 MOTIVATIONS

During the seismic interpretation, the interpreter (usually a geologist or geophysicist) analyzes the seismic volume to find certain geological structures, such as horizons and faults, and hydrocarbon indicators (FIGUEIREDO, 2007). The indicators can be bright spots, which represents, for example, local increase of amplitude followed by phase inversion, usually indicating an accumulation of gas (RAILSBACK 2011). In different lithologies, the occurrences and characteristics of the geological structures in the seismic data will define the geological complexity of the seismic volume (FIGUEIREDO, 2007). For example, complex geological structures like thin layers of the subsurface or multi-plane fault.

Usually, the interpreter starts the seismic interpretation with the identification of geological structures, by tracing horizons and faults in seismic volume in a computational tool. Once faults and horizons are detected, the interpreter manually selects sections (also called subvolumes) of the seismic volume to further investigate the presence of hydrocarbons. Finally, the interpreter proceeds to search for hydrocarbons indicators, such as bright spots, in the seismic sections previously selected. Presented this overview of seismic interpretation and considering the potentially huge size of the seismic volume, manual selection of features, such as searching for bright spots,

requires considerable time and effort by the interpreter. Despite this fact, there is still a lack of a (semi)automatic framework to assist the seismic interpretation in such type of activity (FARFOUR et al. 2012).

To tackle the issues mentioned above, this dissertation aims to propose a framework with a computational process for semi-automatic detection of bright spots, helping interpreters to select relevant seismic sections, focusing only on such sections that potentially have bright spots. Therefore, the interpreter can benefit from the reduction of the search space in the selected seismic section, interpreting and processing only the data in the neighborhood of bright spots that were detected by our framework. Furthermore, to provide a process capable of managing huge seismic data volumes, our framework is implemented in a Massively Parallel Processing Database (MPPDB) infrastructure.

In our framework, we choose bright spots as the seismic pattern to be identified, because such feature is more related to the behavior of the seismic attributes than to the nominal values that change for the same seismic pattern in different seismic datasets. For example, to detect bright spots the search for amplitude increase behavior and phase inversion behavior close to amplitude peak is more important than the nominal values of the amplitude (RAILSBACK 2011). Therefore, our framework considers the behavior of the seismic attributes, producing a more generic and robust solution that works across different seismic datasets.

The proposed framework is underpinned by the Rough Set Theory (PAWLAK 1982). Also, we cogitated to apply Principal Component Analysis (PCA) (JOLLIFFE 2002) to our process. However, previous studies comparing Rough Set with PCA has demonstrated the advantage of the former. PEREIRA et al. (2012) show as the main advantage of using Rough Set the reduction of dependence on human expertise in the decision making. In turn, LEI et. al (2008), demonstrate the overall accuracy of Rough Set as being better than PCA.

About the Rough Set features, there is a formal procedure to eliminate irrelevant conditional attributes within a given information system, yielding to a *reduct*, which contains a subset of conditional attributes able to keep the same properties about all attributes of the same information system. In addition, Rough Set provides a mean to

classify objects of an information system according to the conditional attributes. Besides, the mathematical foundation of Rough Set allows discovering hidden patterns and also provides mechanisms to manage the problem of inaccurate information (such as noisy data) in datasets. Finally, the usefulness of Rough Set to perform data mining and processing signals is proved by a growing number of studies (PAWLAK 1982, 1991; ZARANDI et al. 2008) and, concerning the implementation of Rough Set in seismic dataset we have some papers regarding reservoir prediction (LIU et al. 2007; HONGJIE et al. 2014).

1.2 OBJECTIVE AND HYPOTHESIS

The goal of our research is to assess the applicability of Rough Set to detect potential bright spots in seismic datasets; we also aim to provide a support framework for seismic interpretation. Our proposal is prepared to be implemented in a Massively Parallel Processing Database (MPPDB) infrastructure, aiming to promote faster interpretation for geoscientists. Therefore, we wish to answer the question: In light of Rough Set and MPPDB, how to build a framework to support and speed up seismic interpretation by detecting bright spots? According to this question, we raise the following hypothesis: it is possible to speed up the seismic interpretation process in an MPPDB infrastructure and using Rough Sets for the semi-automatic detection of bright spots.

1.3 DISSERTATION ORGANIZATION

The remainder of this dissertation organized as follows. Section 2 provides the background knowledge of Seismic exploration, MPPDB and Rough Set. Section 3 presents the related works. In Section 4 we present the proposed framework, in Section 5 we discuss our framework design, while Section 6 presents the implementation of our framework. Following in Section 6 we detail the mechanism of our seismic interpretation framework. Section 7 shows the accuracy and performance evaluations of our proposal. Finally, in Section 8 we have the conclusion of our work.

2 BASIC CONCEPTS

This research tackles different areas of knowledge. In this section, we introduce the notions of the seismic exploration, which allow the understanding of the data type used in this study. Also, we explain MPPDB concepts and the Rough Set theory.

2.1 SEISMIC EXPLORATION

The process of searching for oil and gas, using seismic methods, can be divided into three main stages: acquisition, processing, and interpretation (ROBINSON et al. 1980), as depicted in Figure 1 and detailed in next sub-sections.

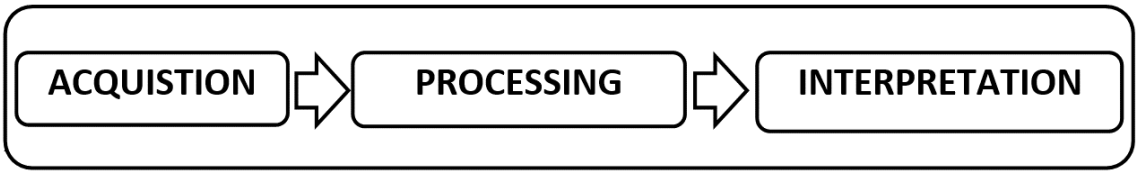


Figure 1. Stages of hydrocarbons exploration based on seismic reflection. Adapted from (SILVA 2004)

2.1.1 Acquisition

The subsoil usually consists of different geological layers with different physical properties. Among these properties, the acoustic impedance is the one that the seismic reflection uses for seismic data acquisition (FIGUEIREDO 2007). This data acquisition is done by generating artificial elastic waves of short duration (order of 200 milliseconds) (THOMAS 2004). This generation of waves occurs at specific points on the surface of the area to be mapped (except in the offshore scenario where we generate the waves on the sea surface, but we want to map the layers from the seabed).

Once generated, the seismic wave propagates in the underground. When it encounters an interface, such as the interface between two types of rock, one part of the wave refracts and propagates into the subsoil. Another part of the wave reflects and returns to the surface where receivers capture it. Such receivers record the arrival time of the wave, and the amount of energy returned. The reflected portion of energy is proportional to the difference between the acoustic impedances two layers in the interface (FIGUEIREDO 2007).

The receivers, located at specific points on the surface, can be (i) geophones – electromagnetic sensors to onshore wave capture or (ii) hydrophones – pressure sensors for data acquisition in marine areas. Once the receivers collect the information, an equipment called seismograph record the data; this device stores the amplitudes of the waves at regular intervals, usually 2 or 4 milliseconds (THOMAS 2004). Figure 2 illustrates such acquisition processes.

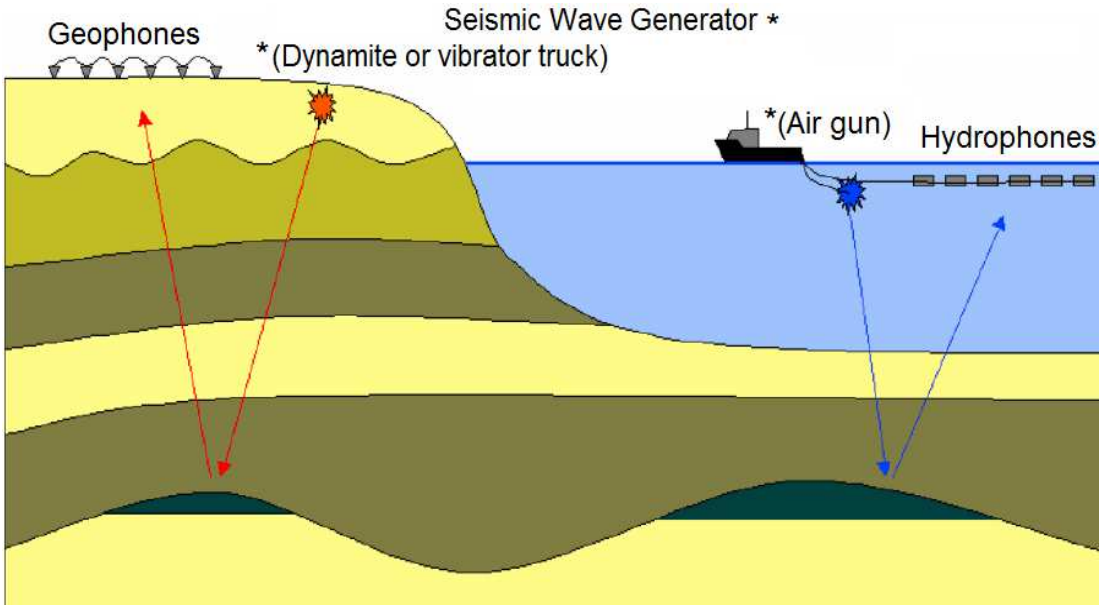


Figure 2. Model of seismic acquisition. Adapted from (GERHARDT 1998)

2.1.2 Processing

In this stage, some errors inherent to the seismic survey, for example, noises are minimized. Furthermore, the data are rearranged to form a three-dimensional grid with seismic amplitude sample at each grid vertex (voxel), presenting two of the dataset dimensions as spatial directions and related to the positions of the sources and seismic receivers. Also, processing stage enables to consider the third dimension of the dataset as the time, and that wave propagation is done only in the vertical direction (SILVA 2004).

For each receiver on the surface, the seismic image obtained will consist of a respective vertical set of amplitudes samples (FIGUEIREDO 2007). This set of samples having the same spatial coordinates, only varying the time, is named as a seismic trace. The maximum and minimum amplitude of the seismic trace are called seismic events.

Figure 3 illustrates the arrangement of samples in the seismic dataset, at the left there is a seismic trace with their amplitudes, undulating signal wave, the only dimension is temporal (1D). At the center (Figure 3), it is a vertical section of the seismic dataset formed by a set of seismic traces, which is named seismic section, with spatial and temporal dimension (2D). For 3D seismic dataset (Figure 3, on the right), we have the seismic volume formed by several seismic sections. In this case, there are two spatial directions, which are inline (parallel to the direction of acquisition) and crossline (perpendicular to the direction of acquisition) plus a temporal direction. Concerning datasets with 2D and 3D, the seismic trace representation presents the seismic attribute color scale (SILVA 2004), sometimes grayscale, where each color represents the intensity of the amplitude value of each sample.

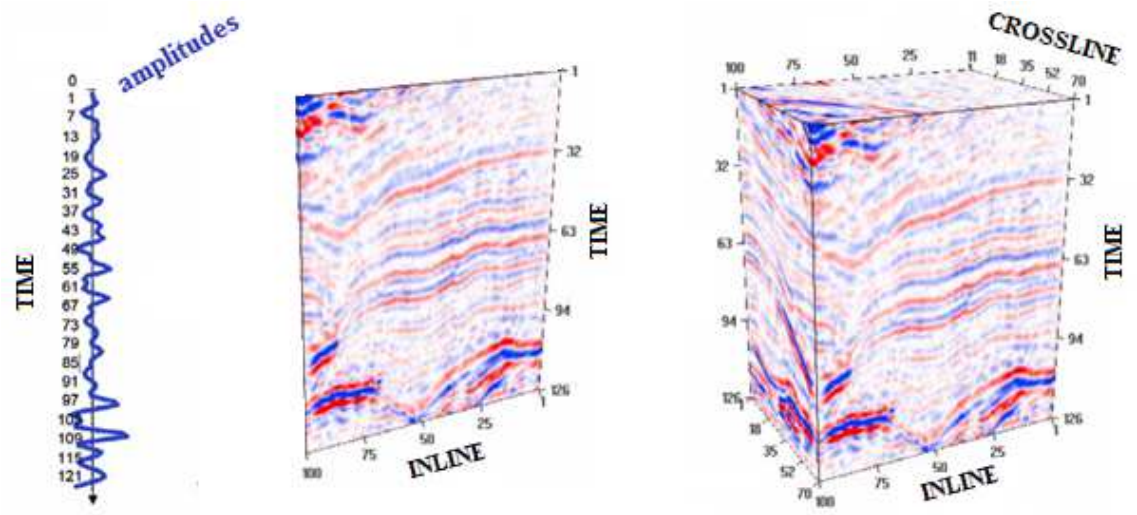


Figure 3. Seismic trace (left), seismic section (center) and seismic volume (right). Adapted from (SILVA 2004)

In our proposed framework, a requirement is that the seismic dataset derives from a SEG-Y that is one of the several standards of seismic file formats developed by the Society of Exploration Geophysicists (SEG) (NORRIS et al. 2002). Also, the SEG-Y must be post-stack file format (SEG-Y pre-processed), instead of SEG-Y pre-stack file (SEG-Y without pre-processing).

2.1.3 Interpretation

In the interpretation stage, the interpreter analyzes the seismic dataset and attempts to create a model that represents the geology of the survey area. Figure 4 shows a geological model that could be the result when interpreting a seismic section.

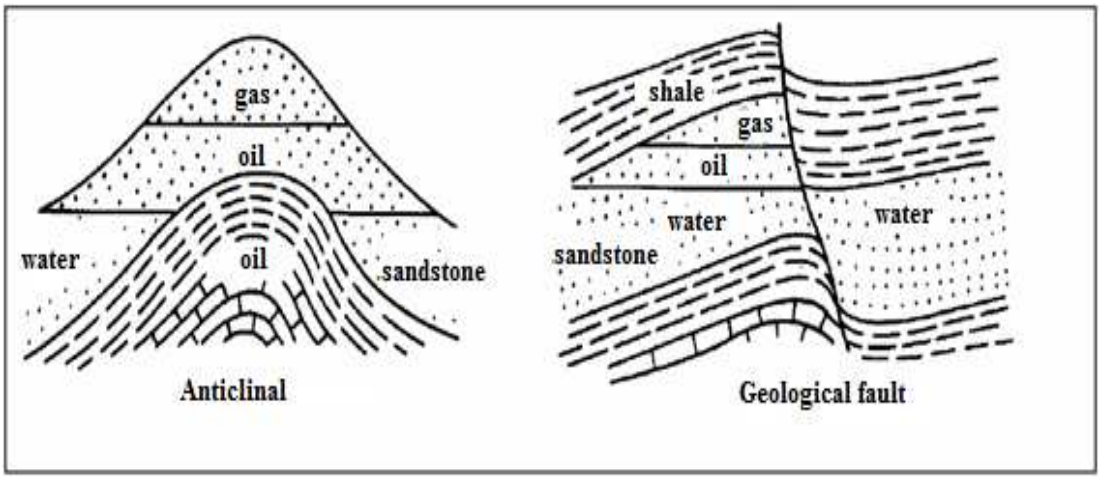


Figure 4. Geological model. Adapted from (ROBINSON et al. 1980)

In Figure 4, we can see the representation of the subsoil layers, where the interface that separate two different layers is called seismic horizon. Such interface associates with a reflection stretching for an enormous area (SHERIFF 1991) and, it is detected by a series of continuous reflections of similar intensities found in the lateral vicinity along the seismic dataset.

Advances in data acquisition, processing and interpretation now make it possible to use seismic traces to reveal more than just shape and position of the reflector. Changes in the character of seismic pulses returning from the reflector can be interpreted to verify the depositional history of the basin, the rock type in a layer, and even the nature of the pore fluid. This last refinement (pore fluid identification) can be reached using the attribute energy from a trace segment (DGB, 2015b). Equation 1 shows how to calculate such trace segment energy (*E)*:

$$E = \frac{(\sum_{i=ts}^{te} s_i^2)}{n} \qquad (1)$$

Where, *ts* is the time start and *te* is the time end for the desired trace segment; *s* is the sample amplitude value, and *n* is the number of samples in the time interval.

This attribute can enhance seismic events and, therefore, useful to detect a seismic pattern, such as bright spots (Figure 5), since the response energy characterizes the acoustic rock properties (DGB, 2015b).

Normally, in the pore fluid detection, also it is possible to perform AVO (Amplitude Versus Offset) analysis (CHIBURIS et al. 1993) but preferable for pre-stack SEG-Y. Since in our approach we envision to attend SEG-Y post-stack data, the use of AVO would reflect negatively on our accuracy (ANDERSON et al. 2009).

The early practical evidence that seismic waves could detect fluids came from bright spots, which often signify gas accumulation. Since the early 1970s, bright spots have been the goal of seismic exploration (GEORGE 1997); they comprise amplitude events that indicate the presence of hydrocarbons. Despite other improvements in seismic acquisition and interpretation, the bright spots as hydrocarbon indicators continue to be sought, even as data density and increased resolution vastly expand the volume of the data itself. As a consequence, the interpreter is confronted by a massive amount of data from which he/she must quickly derive an accurate evaluation.

One of the main objectives of seismic interpretation is to evaluate the chance of seismic events that may be related to the presence of hydrocarbons (RODEN et al., 2005). The seismic events in our study conform to bright spots associated with hydrocarbons, Figure 5 depicts such hydrocarbons indicators.

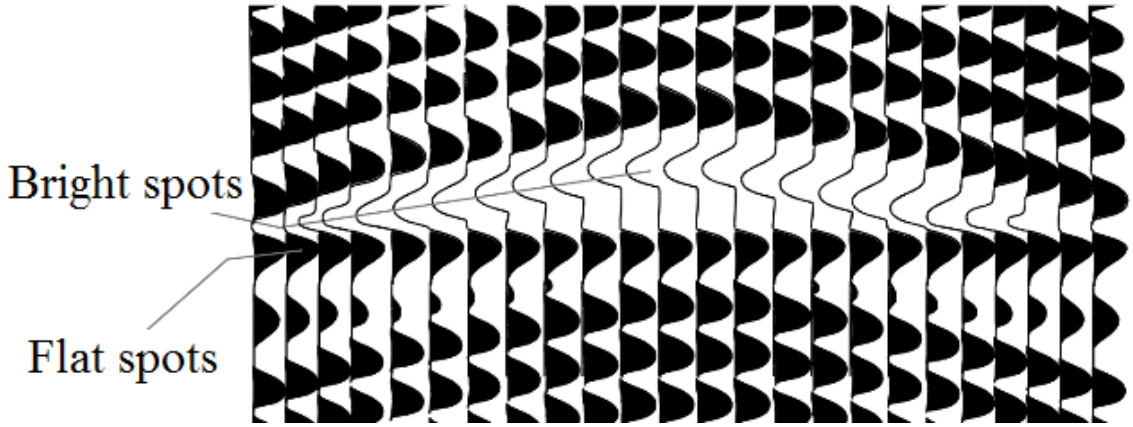


Figure 5. Hydrocarbon indicators. Bright spots and flat spots in the seismic dataset. Adapted from (RAILSBACK 2011).

In a hydrocarbon indicator of type bright spots, the gas presence in the horizon pores causes a dramatic decrease in acoustic impedance compared to the encasing shale (high amplitude value). Also, it is remarkable a phase inversion for the amplitudes of a seismic trace in the bright spots region, as noticed in Figure 5. Such typical bright spots

seismic events are identified if appropriate seismic attributes have been used (FARFOUR et al. 2012).

The flat spots feature, a particular case of bright spots, occurs by the contrast between the small acoustic impedance of gas-filled porous rock and the greater acoustic impedance of liquid-filled porous rock, at horizontal gas-liquid contact. Flat spots feature is recognizable because it is discordant with the non-horizontal surrounding structures as we can see in Figure 5 (RAILSBACK 2011).

2.2 MPPDB INFRASTRUCTURE

In the present research, we adopted as MPPDB infrastructure the Pivotal Greenplum (GOLLAPUDI 2013; PIVOTAL 2015), which is based on PostgreSQL. Also, Greenplum is a representative example of MPPDB, it is Open Source, and it is based on the shared-nothing MPP architecture. Figure 6 shows this architecture where data is partitioned across multiple segment servers.

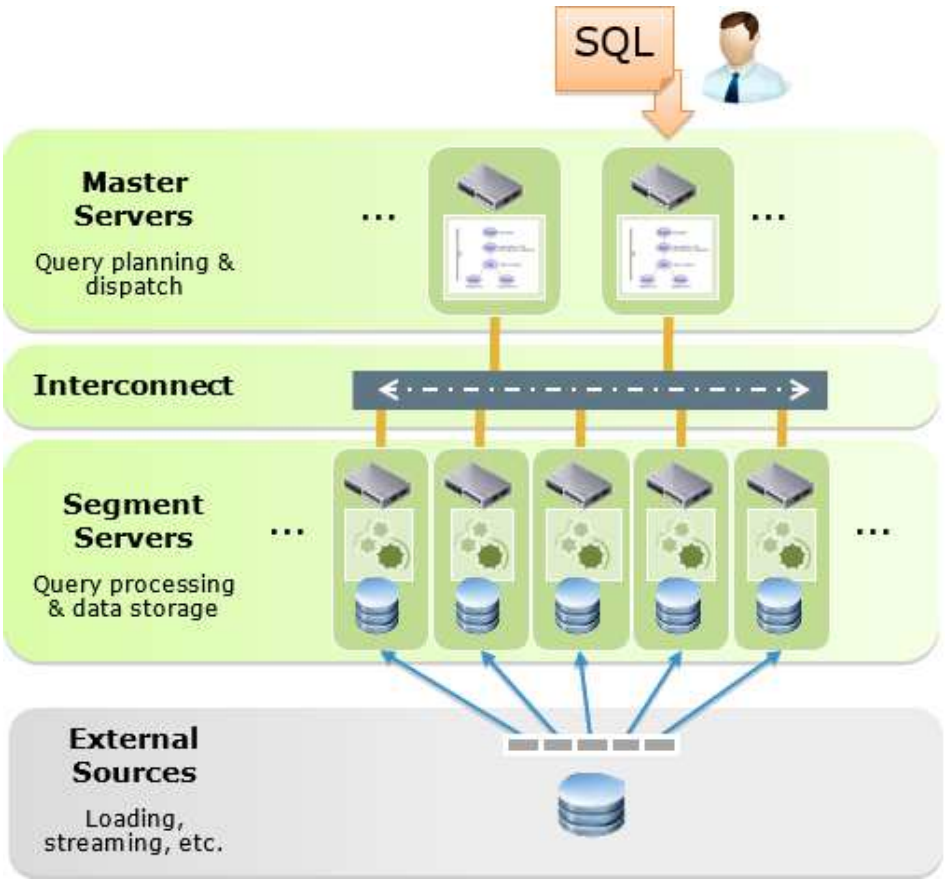


Figure 6. Greenplum shared-nothing massively parallel processing architecture. Adapted from (PIVOTAL 2015)

The master server is the entry point to the Greenplum database system. The architecture supports backup of the master server to be used when the primary master server becomes non-operational. The segment servers are where data is stored, and the majority of the processes are created to handle the work of a query. Each segment server has its processor, memory, disk, operating system and manages a distinct portion of the overall data (PIVOTAL 2015).

The shared-nothing architecture presents challenges to designers of database schemas. In particular, the correct allocation of data in the segments is an important point of analysis, which can directly affect the efficiency and scalability. The configuration of the data distribution is done through the Data Definition Language (DDL) commands extended to allow the specification of the distribution policies of the data in the segments (PIVOTAL 2015). For example, the following command creates a table using *O_ORDERKEY* column as a distribution key. Thus, the table rows are assigned to segments in accordance with the value of *O_ORDERKEY* column:

CREATE TABLE *ORDERS* (*O_ORDERKEY* INT, ...)
DISTRIBUTED BY (*O_ORDERKEY*)

The configuration of the distribution key should ideally ensure that data, commonly requested by the same query, are grouped into the same segment. If such data is stored in different segments, during queries execution, the data motion occurs between segments, which characterizes a *processing skew* since the query processing time is increased (PIVOTAL 2015). Figure 7 shows *processing skew* scenario with data motion.

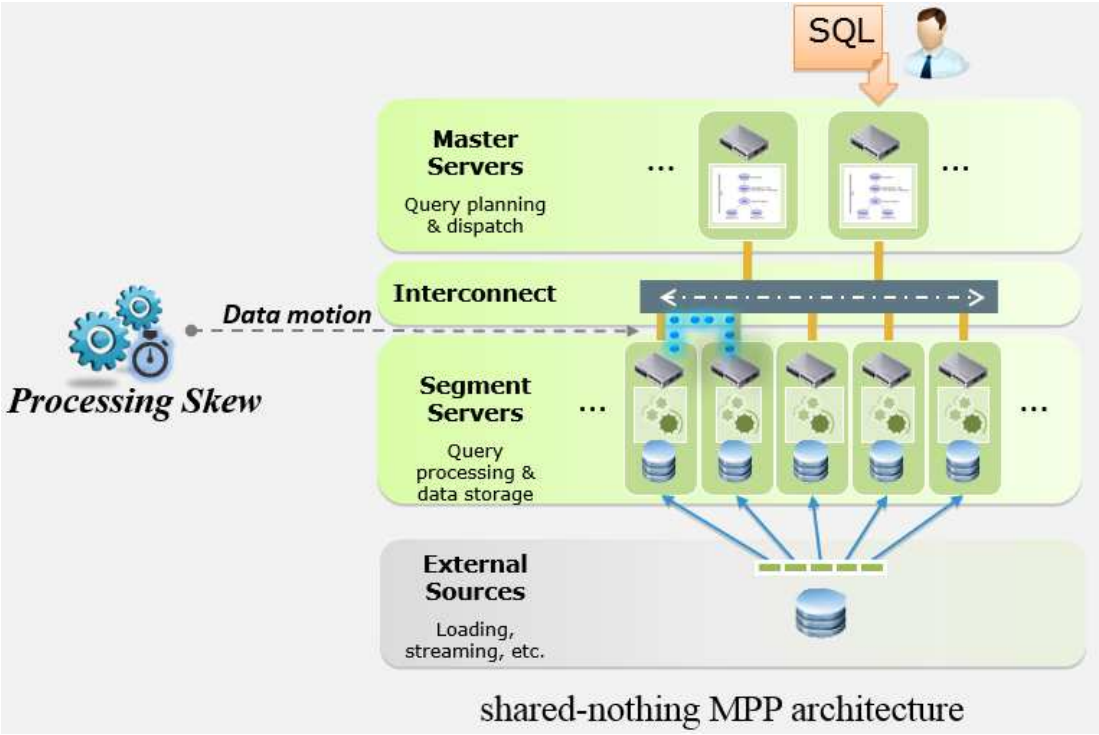


Figure 7. Data motion and processing skew scenario. Adapted from (PIVOTAL 2015)

Furthermore, the distribution key configuration is directly associated with the segment's data balance (containing approximately the same amount of data). Otherwise, an unbalanced data allocation, which is called *data skew* (PIVOTAL 2015), can negatively impact the performance. Figure 8 depicts the *data skew* scenario yielding a *processing skew* when during queries execution the segment with the larger volume of information can become a bottleneck in the process.

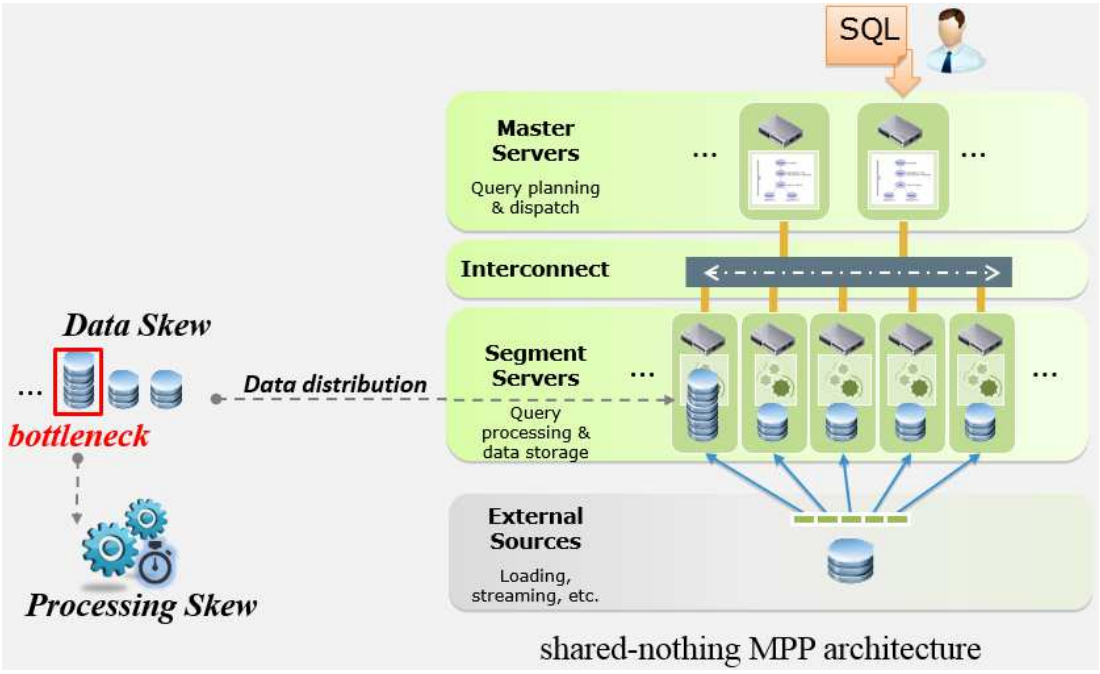


Figure 8. Data skew and processing skew scenarios. Adapted from (PIVOTAL 2015)

2.3 ROUGH SET THEORY

Rough Set based data analysis usually starts with a data table, called information system. The information system comprises data about objects of interest characterized regarding some conditional attributes and presenting a decision attribute, for this reason, such information system is also called a decision table that describes decisions in terms of conditions. With every decision table a set of classification rules can be associated (PAWLAK 2002).

Information systems containing data redundancy are more costly to be processed, in this context, Rough Set Theory presents a process to eliminate such data redundancy. Rough Set theory can also be used to classify objects of an information system, according to the object's conditional attributes, and if there is inaccurate information, Rough Set can manage such imprecisions, noisy and incomplete information present in the information system. Thus, objects that cannot be specified by the available data will be classified in this theory through concepts of lower and upper approximations (KOMOROWSKI et al. 1999; PATRÍCIO et al. 2005).

Considering the above, in the methodology to design and implement our framework, we will creates an information system and apply Rough Set on it aiming: (i) to reduce the number of seismic attributes, keeping only the relevant ones to bright spots detection; (ii) to generate classification rules by a supervised training process, rules for the pattern recognition process to detect bright spots, and (iii) incorporates specialist's knowledge in our framework to improve the accuracy of our support process.

2.3.1 Information system

On Rough Set approach, a common way for data representation is via an information system (*IS*), which can be defined by the Equation 2.

$$IS = (U, C \cup D) \qquad (2)$$

Where, U is a set of objects, where each object has some conditional attributes (C) and, a decision attribute (D) that classifies objects according to some criteria as the example in Table 1. Such attributes are the same for each of the objects in *IS*, but their nominal values may differ (KOMOROWSKI et al. 1999; PATRÍCIO et al. 2005).

Table 1 Information system example. Adapted from (KOMOROWSKI et al. 1999).

U	**C**					**D**
Toy	**Color**	**Size**	**Touch**	**Texture**	**Material**	**Child's reaction**
1	blue	big	hard	undefined	plastic	negative
2	red	medium	moderate	flat	wood	neutral
3	yellow	small	soft	rugged	plush	positive
4	blue	medium	moderate	rugged	plastic	negative
5	yellow	small	soft	undefined	plastic	neutral
6	green	big	hard	flat	wood	positive
7	yellow	small	hard	undefined	metal	positive
8	yellow	small	hard	undefined	plastic	positive
9	green	big	hard	flat	wood	neutral
10	green	medium	moderate	flat	plastic	neutral

Table 2 Attributes nominal values. Adapted from (KOMOROWSKI et al. 1999).

	Attribute	Nominal Values
C	Color	blue, red, yellow, green
	Size	big, medium, small
	Touch	hard, moderate, soft
	Texture	flat, rugged, undefined
	Material	plastic, wood, plush, metal
D	Attitude	neutral, negative, positive

2.3.2 Discernibility matrix and discernibility function

Be an equivalence class (*Cl*) determined by a set of nominal values of *C* that exist at least for one object in *IS*. Considering *IS* and *C*, we have a discernibility matrix denoted $M_d(C)$ symmetric $n \times n$, where n equals the number of existing equivalence classes in *IS*. Thus, a discernibility matrix element $M_d(i, j)$, where $i, j = 1, ..., n$, it is a set of conditional attributes $B \subseteq C$ that differentiates objects from two equivalence classes (KOMOROWSKI et al. 1999; PATRÍCIO et al. 2005). Table 3 depicts an example of discernibility matrix, where in a simplest representation we have Color equals Co; Size equals Sz; Touch equals To; Texture equals Te; Material equals Ma.

Table 3 Discernibility matrix. Adapted from (KOMOROWSKI et al. 1999).

	Cl1	Cl2	Cl3	Cl4	Cl5	Cl6	Cl7	Cl8	Cl9
Cl1	Ø								
Cl2	Co,Sz,To, Te,Ma	Ø							
Cl3	Co,Sz,To, Te, Ma	Co,Sz,To, Te,Ma	Ø						
Cl4	Sz,To,Te	Co,Te,Ma	Co,Sz,To, Ma	Ø					
Cl5	Co,Sz,To	Co,Sz,To, Te,Ma	Te,Ma	Co,Sz,To, Te	Ø				
Cl6	Co,Te,Ma	Co,Sz,To	Co,Sz,To, Te, Ma	Co,Sz,To, Te, Ma	Co,Sz,To, Te, Ma	Ø			
Cl7	Co,Sz,Ma	Co,Sz,To, Te,Ma	To,Te,Ma	Co,Sz,To, Te, Ma	To,Ma	Co,Sz,Te, Ma	Ø		
Cl8	Co,Sz	Co,Sz,To, Te,Ma	To,Te,Ma	Co,Sz,To, Te	To	Co,Sz,Te, Ma	Ma	Ø	
Cl9	Co,Sz,To, Te	Co,Ma	Co,Sz,To, Te, Ma	Co,Te	Co,Sz,To, Te	Sz,To,Ma	Co,Sz,To ,Te,Ma	Co,Sz,To, Te	Ø

The discernibility function $F_d(C)$ is a Boolean function that determines the minimum set of attributes from C to differentiate any equivalence class of the others. $F_d(C)$ is obtained as follows: for the attributes contained within each discernibility matrix element, one applies the operator *sum* or *or* or ∨ and, among the discernibility matrix elements, one uses the operator *product* or *and* or ∧, resulting in a Boolean expression of *Product of Sum*. The $F_d(C)$ of the $M_d(C)$ in Table 3 is represented by the Equation 3 (KOMOROWSKI et al. 1999):

$$
\begin{aligned}
F_d(C) = {} & (Co \vee Sz \vee To \vee Te \vee Ma) \wedge (Co \vee Sz \vee To \vee Te \vee Ma) \wedge (Sz \vee To \vee Te) \\
& \wedge (Co \vee Sz \vee To) \wedge (Co \vee Te \vee Ma) \wedge (Co \vee Sz \vee Ma) \wedge (Co \vee Sz) \wedge (Co \vee \\
& Sz \vee To \vee Te) \wedge (Co \vee Sz \vee To \vee Te \vee Ma) \wedge (Co \vee Te \vee Ma) \wedge (Co \vee Sz \vee \\
& To \vee Te \vee Ma) \wedge (Co \vee Sz \vee To) \wedge (Co \vee Sz \vee To \vee Te \vee Ma) \wedge (Co \vee Sz \vee \\
& To \vee Te \vee Ma) \wedge (Co \vee Ma) \wedge (Co \vee Sz \vee To \vee Ma) \wedge (Te \vee Ma) \wedge (Co \vee Sz \\
& \vee To \vee Te \vee Ma) \wedge (To \vee Te \vee Ma) \wedge (To \vee Te \vee Ma) \wedge (Co \vee Sz \vee To \vee Te \\
& \vee Ma) \wedge (Co \vee Sz \vee To \vee Te) \wedge (Co \vee Sz \vee To \vee Te \vee Ma) \wedge (Co \vee Sz \vee To \\
& \vee Te \vee Ma) \wedge (Co \vee Sz \vee To \vee Te) \wedge (Co \vee Te) \wedge (Co \vee Sz \vee To \vee Te \vee Ma) \\
& \wedge (To \vee Ma) \wedge (To) \wedge (Co \vee Sz \vee To \vee Te) \wedge (Co \vee Sz \vee Te \vee Ma) \wedge (Co \vee \\
& Sz \vee Te \vee Ma) \wedge (Sz \vee To \vee Ma) \wedge (Ma) \wedge (Co \vee Sz \vee To \vee Te \vee Ma) \wedge (Co \\
& \vee Sz \vee To \vee Te)
\end{aligned}
\tag{3}
$$

To conclude, using theorems, properties, and postulates of Boolean algebra, it is possible to get the minimized expression (KOMOROWSKI et al. 1999; PATRÍCIO et al. 2005). The minimized expression for our $F_d(C)$ example is represented by the Equation 4:

$$F_d(C) = ((Co \wedge To \wedge Ma) \vee (Sz \wedge To \wedge Te \wedge Ma)) \tag{4}$$

Finally, using the discernibility matrix and discernibility function, the reduction of attributes and generation of classification rules can be done (PAL et al. 1999; VASHIST et al. 2011) (See Section 2.3.3).

2.3.3 Reduction of the information system and generation of classification rules

In an information system, there is usually among the stored attributes the ones that are relevant, and some that are irrelevant to a particular decision-making. For example, some attributes from a seismic dataset may not be pertinent to identify a horizon on a seismic interpretation. Thus, for a process to assist in a specific decision, only a subset

of attributes is necessary. Rough Set theory has a procedure to eliminate irrelevant attributes within an information system. This procedure outcome is a *reduct*, which determines a set of minimum attributes needed to preserve the knowledge of an information system having all their attributes. Consequently, the *reduct* can classify objects without affecting the knowledge representation (PAWLAK 1982). If such reduction process does not find any information system *reduct*, it means that all the attributes in C are relevant, in this case, all the *IS* conditional attributes are kept, and *reduct* equals C.

Summing up, an information system *reduct* is a set of attributes R where $R \subseteq C$, being dispensable all the attributes $d \in (C - R)$, where the minimum terms of discernibility function $F_d(C)$ determine the *reducts* of C (see Equation 4). Therefore in an information system, we may find multiple *reducts* (KOMOROWSKI et al. 1999; PATRÍCIO et al. 2005). Regarding our example, in Equation 4 we can identify the *reducts* of our *IS.* Table 4 shows one of such *reducts*.

Table 4 *Reduct* example. Adapted from (KOMOROWSKI et al. 1999).

U	**R**		
Toy	**Color**	**Touch**	**Material**
1	blue	hard	plastic
2	red	moderate	wood
3	yellow	soft	plush
4	blue	moderate	plastic
5	yellow	soft	plastic
6	green	hard	wood
7	yellow	hard	metal
8	yellow	hard	plastic
9	green	hard	wood
10	green	moderate	plastic

Following the classification rules are built using the attributes R and such rules can be expressed in the form:

IF *EXP* THEN *D*

Where *EXP* is a conjunctive expression whose clauses are R with their nominal values and D with its nominal value. Considering our information system *reduct* example (Table 4), we have the rules in Table 5:

Table 5 Classification rules example. Adapted from (KOMOROWSKI et al. 1999).

Rules	R1: IF	Color = blue	AND	Touch = hard	AND	Material = plastic	THEN	Child's reaction = negative
	R2: IF	Color = red	AND	Touch = moderate	AND	Material = wood	THEN	Attitude = neutral
	R3: IF	Color = yellow	AND	Touch = soft	AND	Material = plush	THEN	Attitude = positive
	R4: IF	Color = blue	AND	Touch = moderate	AND	Material = plastic	THEN	Attitude = negative
	R5: IF	Color = yellow	AND	Touch = soft	AND	Material = plastic	THEN	Attitude = neutral
	R6: IF	Color = green	AND	Touch = hard	AND	Material = wood	THEN	Attitude = positive
	R7: IF	Color = yellow	AND	Touch = hard	AND	Material = metal	THEN	Attitude = positive
	R8: IF	Color = yellow	AND	Touch = hard	AND	Material = plastic	THEN	Attitude = positive
	R9: IF	Color = green	AND	Touch = hard	AND	Material = wood	THEN	Attitude = neutral
	R10: IF	Color = green	AND	Touch = moderate	AND	Material = plastic	THEN	Attitude = neutral

The rules are called non-deterministic or inconsistent when they share the same nominal values for conditional attributes but have different nominal values for the decision attribute, for example, the rules *R6* and *R9* in Table 5. By applying non-deterministic rules, one cannot affirm that the decision-making will be correct. In addition, the rules are said deterministic or consistent when they share the same values for both conditional attributes and the decision attribute (KOMOROWSKI et al. 1999; PATRÍCIO et al. 2005), for example the rules *R1*, *R2*, *R3*, *R4*, *R5*, *R7*, *R8* and, *R10* in Table 5.

3 RELATED WORKS

In the literature, studies related to seismic interpretation with pattern recognition comprise three groups of proposals: (i) detection of horizons and/or faults (BAKKE et al. 2012; SONG et al. 2012; YU et al. 2013; BASIR et al. 2013); (ii) Reservoir prediction (HERRERA et al. 2006; LIU et al. 2007; CLIFFORD et al. 2011; HONGJIE et al. 2014); and (iii) detection of hydrocarbon indicators (FARFOUR et al. 2012)

Concerning the detection of horizons and faults, BAKKE et al. (2012) propose a technique called seismic DNA that translates the continuous seismic data into characters. Once the translation is done, the next step is to create a regular expression which describes the feature the interpreter is searching, allowing the interpreter do the translation and the design of the search expression using a graphical user interface to interact with the seismic data. They demonstrate their method with the extraction of a horizon from a seismic volume with the additional condition that such horizon should be intersected by faults. About the research of SONG et al. (2012), they present a method for fault detection; such method is based on surface fitting algorithm, which is a popular method used for image edge detection. The dataset adopted to validate their technique is part of Netherlands offshore F3 volume. With reference to YU et al. (2013), they propose a pattern recognition-based algorithm for horizon auto-tracking, generating orientation vectors from seismic amplitude data and guide the pick selection. In addition, they apply a minimum-spanning tree (MST) algorithm to guide and optimize the trace selection, which yields a complete and accurate horizon. Lastly, Basir et al. propose a fault detection, pre- processing the seismic dataset, then computing the seismic attributes and, applying the techniques ant-tracking and an unsupervised artificial neural network (ANN) to combine multiple attributes to achieve the fault detection.

Regarding reservoir prediction HERRERA et al. (2006) and, CLIFFORD et al. (2011) both proposals apply ANN aiming reservoir characterization, and HERRERA et al. (2006) generate seismic attributes that are related to the reservoir properties and combining these attributes to predict the properties of the reservoir. LIU et al. (2007) and, HONGJIE et al. (2014) they address Rough Set as the main process to reduce the seismic attributes, showing that attribute reduction not only can satisfy the prediction

precision but also can save cost, improve process speed and have a remarkable effect on oil-gas prediction.

Finally, the work of FARFOUR et al. (2012) is the most similar to our proposed framework. Such paper proposes a method to recognize bright spots associated with hydrocarbon accumulation. Their method combines and transforms seismic attributes to detect seismic patterns, using an ANN with a supervised training, incorporating specialist's knowledge. Also, they use the seismic dataset F3 to demonstrate their approach. Despite having the same goal as our work, FARFOUR et al. (2012) do not even test their proposal for performance evaluation and, they do not mention any requirement of efficiently managing a large amount of data, which is one of our requirements. Moreover, unlike the research of FARFOUR et al., our Rough Set based process comprises generic rules for pattern recognition, providing more robustness to our framework that does not need to generate new classification rules for different seismic datasets of study, as it is the case of their approach that trains the ANN for each different seismic dataset, when the interpreter decides the seismic pattern to be addressed and picks examples for the ANN.

4 PROPOSED FRAMEWORK

Our proposal consists of a seismic interpretation support framework for semi-automatic detection of bright spots, using the Rough Set theory and an MPPDB infrastructure. An interpreter uses the proposed framework to run pattern recognition queries in a given seismic dataset to have the indication of bright spots, reducing the search space and thus reducing the time spent on the seismic interpretation activity.

In Figure 9, we can see the steps of seismic interpretation until the interpreter achieves all the bright spots associated with hydrocarbon accumulation, considering two scenarios: (i) on the left, we depict the usual steps followed by the interpreter; we can notice that the selection of seismic sections is manual and the interpreter would process all the data from selected seismic sections; (ii) on the right, we describe the steps when the interpreter uses our framework; we can see that the selection of seismic sections is semi-automated and there is no need to analyze all data from selected seismic sections.

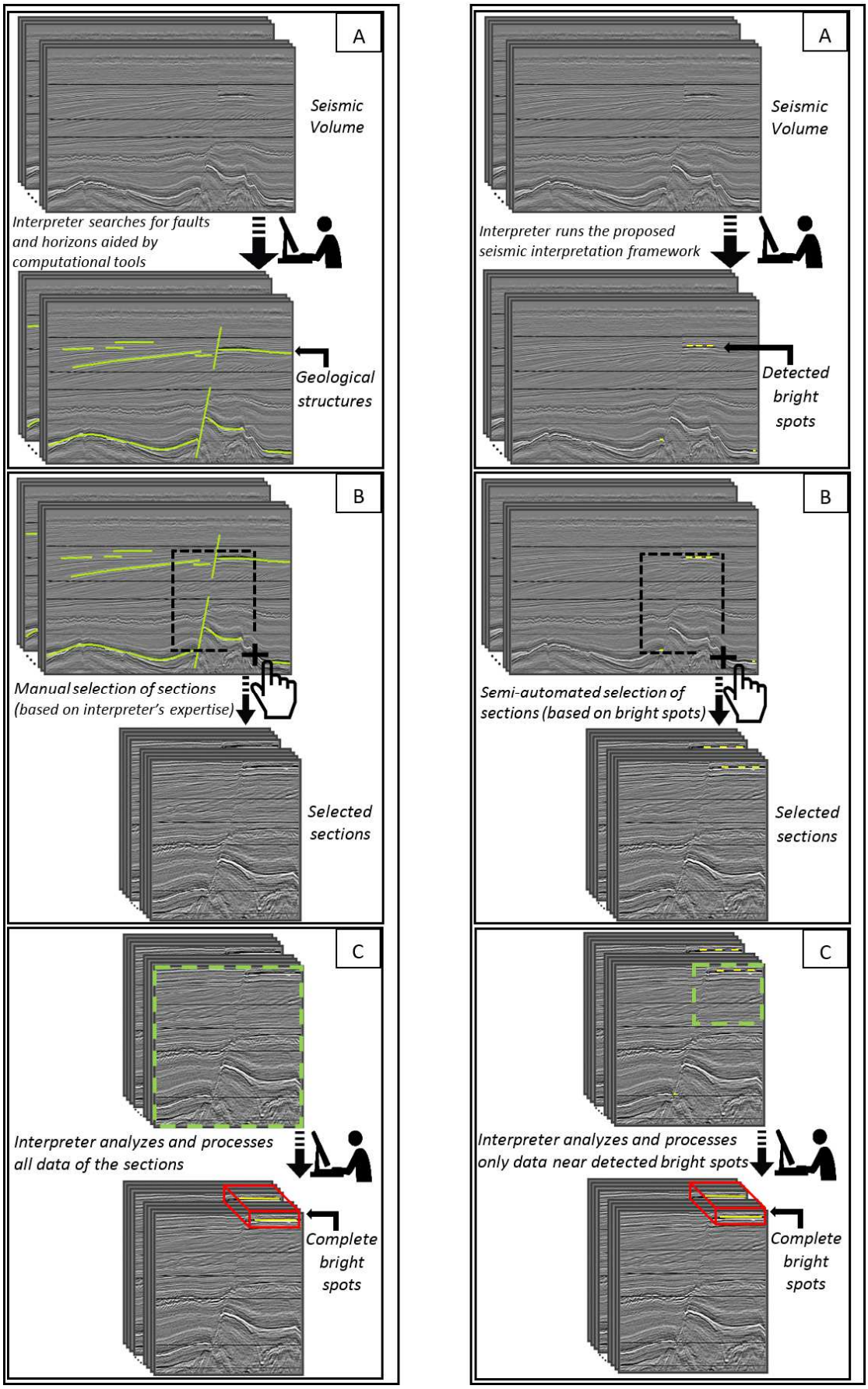


Figure 9. Seismic Interpretation to detect bright spots: on the left, following usual steps; on the right, using our framework

5 FRAMEWORK DESIGN

In this section, we conceptually explain the methodology to design our framework in light of Rough Set and seismic interpretation aspects. The following subsections will address the planning for our information system, attributes reduction and rules generation.

5.1 INFORMATION SYSTEM DESIGN

To design our information system, we analyzed a seismic section of a well-known real seismic volume, which was already interpreted as containing bright spots associated with hydrocarbon accumulation. The chosen seismic volume, called F3, is from the gas reservoir of the Dutch coast and is available on dGB Earth Sciences web page (DGB, 2015a) in a SEG-Y file format.

The object of our information system will be the seismic trace. Thus, to design our Information System, it is necessary to define *C* attributes and *D* (as discussed in Section 2.3.1), where *C* must be related to our *D* that states the presence of bright spots, which are the seismic pattern the interpreter wishes to detect. Thus, we will conveniently define our *C* to highlight bright spots in light of a specialist's knowledge in seismic interpretation, the conceptual definition of the bright spots and the interpreted seismic volume F3.

Regarding literature on seismic attributes, HAMPSON et al. (2001) defined seismic attributes as any mathematical transformation of the seismic trace data, which goes from simple attributes such as amplitude, phase, and frequency, to complex attributes such as AVO. Moreover, such transformations could incorporate other data sources than the seismic dataset itself. In our proposal, to define our seismic conditional attributes we will follow the Hampson's definition, extracting from a SEG-Y file the seismic trace data. Still, on the generation of our seismic conditional attributes, we will incorporate on such attributes the information from a neighborhood of the seismic trace, which is a relevant requirement for the pattern recognition process, since our pattern associates with a region of seismic traces. Thus, in our framework we will have the input parameter *trace_search_radius* (Figure 10), defining the neighborhood size (window) in terms of the number of consecutive seismic traces inside the region to be

analyzed. Therefore such parameter provides the expected horizontal size for the pattern bright spots (see Section 6 and Section 7).

Also, for our seismic conditional attributes, since we are interested in amplitude peaks we have to know if the phase reference is positive or negative for the amplitudes of seismic traces, such reference is inferred from the analysis of the SEG-Y file by a specialist. If we have a positive phase reference, then the maximum amplitudes of seismic traces will present positive values; therefore negative phase reference implicates negative values for maximum amplitudes of seismic traces. The SEG-Y F3 that we will use in our framework implementation has positive phase reference. However we must prepare our framework to handle both values for phase reference, therefore to be more robust, providing to the final user of our framework a phase reference input parameter.

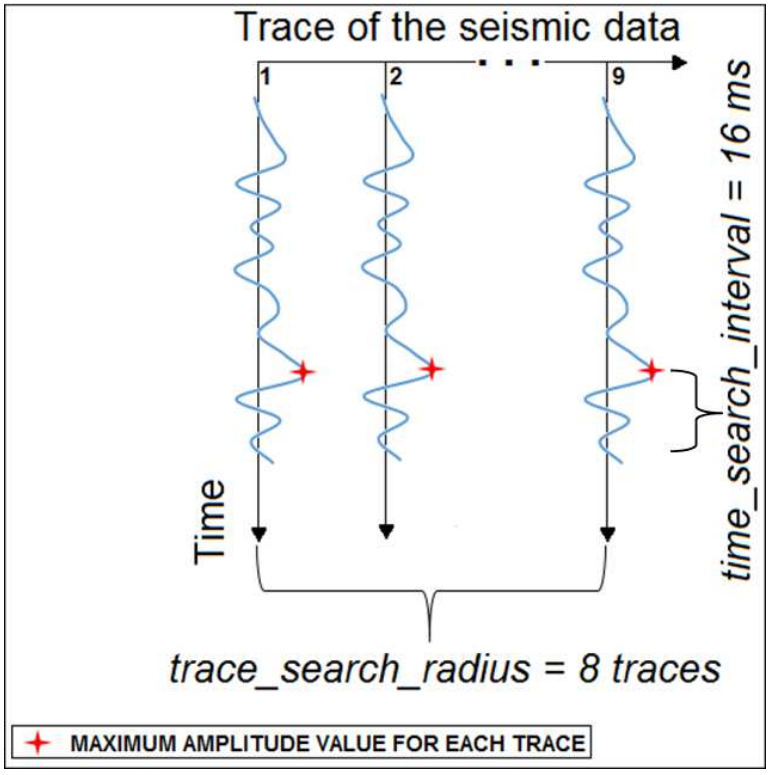


Figure 10. Schematic example for the input parameters trace_search_radius (e.g. value 8 traces in the neighborhood of an object seismic trace) and time_search_interval (e.g. value 16 ms)

Concerning the seismic dataset as a source to design our information system, only one seismic dataset with bright spots will be enough. Because instead of generating our conditional attributes based on nominal values of attributes within the seismic trace, which change from one seismic dataset to other, our conditional attributes will consider

the behaviors of such attributes close to bright spots, and the behaviors are the same for different seismic datasets with bright spots. Thus we designed a set of generic seismic conditional attributes for our framework, providing a generic pattern recognition process, where we will look for amplitude phase inversion that is typical of a bright spots region (see Section 6 and Section 7) that counts only on attributes behaviors to detect bright spots independently of the seismic dataset. Regarding the discretization of our conditional attributes values, we defined intervals that are related to the behavior of each trace in the neighborhood of the current seismic trace.

A specialist geologist helped us to choose the discrete values for our seismic conditional attributes empirically. Also, if necessary, the final user will be able to adjust the attributes discretization indirectly, since values of attributes are linked to the input parameters *trace_search_radius* and *time_search_interval* (Figure 10). This *time_search_interval* defines the expected vertical size of the pattern bright spots, which is an interval time in milliseconds starting on the higher amplitude within a seismic trace (See Section 6 and Section 7). Following, we present the conditional attributes defined for our information system.

- **bright_spot_pieces_region**

  This conditional attribute means the continuity/frequency of the bright spots traces in the trace neighborhood, where some traces may be a piece of bright spots and others may not, representing discontinuities. The best candidate for the bright spots is the one that comprises a neighborhood with 100% of continuity; it means that the neighborhood is compounded only by traces that were classified as bright spots piece. Concerning the discrete values of *bright_spot_pieces_region* they are according to the percentage of traces classified as bright spots piece within the neighborhood, as following:

  - *'>90%TraceRadius'* when the percentage is greater than or equal to 90.
  - *'[80-90[%TraceRadius'* when the percentage is greater than or equal to 80 and less than 90.
  - *'[70-80[%TraceRadius'* when the percentage is greater than or equal to 70 and less than 80.
  - *'[60-70[%TraceRadius'* when the percentage is greater than or equal to 60 and less than 70.
  - *'<60%TraceRadius'* when the percentage is less than 60.

- **sample_max_indexes_region**

  The present conditional attribute represents the horizontality of the bright spots region, such horizontality is related to the position (index) of the higher amplitude sample (*sample_max*) of each seismic trace in the trace neighborhood, indicating how well aligned are such samples. The higher horizontality occurs when in the neighborhood all the indexes of *sample_max* are the same. The best bright spots candidate has the best horizontal alignment. Concerning the discrete values of *sample_max_indexes_region* they are according to the difference in samples between the highest index and the lowest index among all the *sample_max* of the traces within the neighborhood, as following:

  - *'Inclination0Sample'* when the difference is zero.
  - *'Inclination1Sample'* when the difference is one sample.
  - *'Inclination2Samples'* when the difference is two samples.
  - *'Inclination3Samples'* when the difference is three samples.
  - *'Inclination>4Samples'* when the difference greater than or equal to four samples.

The discrete values of our conditional attributes will compose the clauses of the conjunctive expression in the classification rules.

5.2 REDUCT AND CLASSIFICATION RULES DESIGN

To extract *reducts* and classification rules from our information system, we will employ Rosetta (ØHRN 2001), which is a general-purpose toolkit for analyzing tabular data in light of Rough Set discernibility-based model. Such toolkit is designed to support the overall data mining and knowledge discovery process via computation of minimal attributes sets and generation of classification rules in a supervised training process, using the concepts of discernibility matrix and discernibility function (PAWLAK 1982; SELVI et al. 2014). Such training is supervised because we will indicate the objects that are bright spots, based on the SEG-Y F3 already interpreted by a specialist. Therefore we will incorporate the specialist's knowledge in our proposal framework (see Section 6).

6 FRAMEWORK IMPLEMENTATION

Concerning the implementation of our Framework, the UML activity diagram depicted in Figure 11 provides an overview of the process followed in the implementation, comprising two sub-processes: (i) *Information System and Classification Rules Specification* (ii) *Rough Sets*. The developer of our framework is responsible for executing the framework implementation activities, but the final interpreter user of our framework is not in charge to run such implementation activities. Next, we describe in detail each activity with their artefacts.

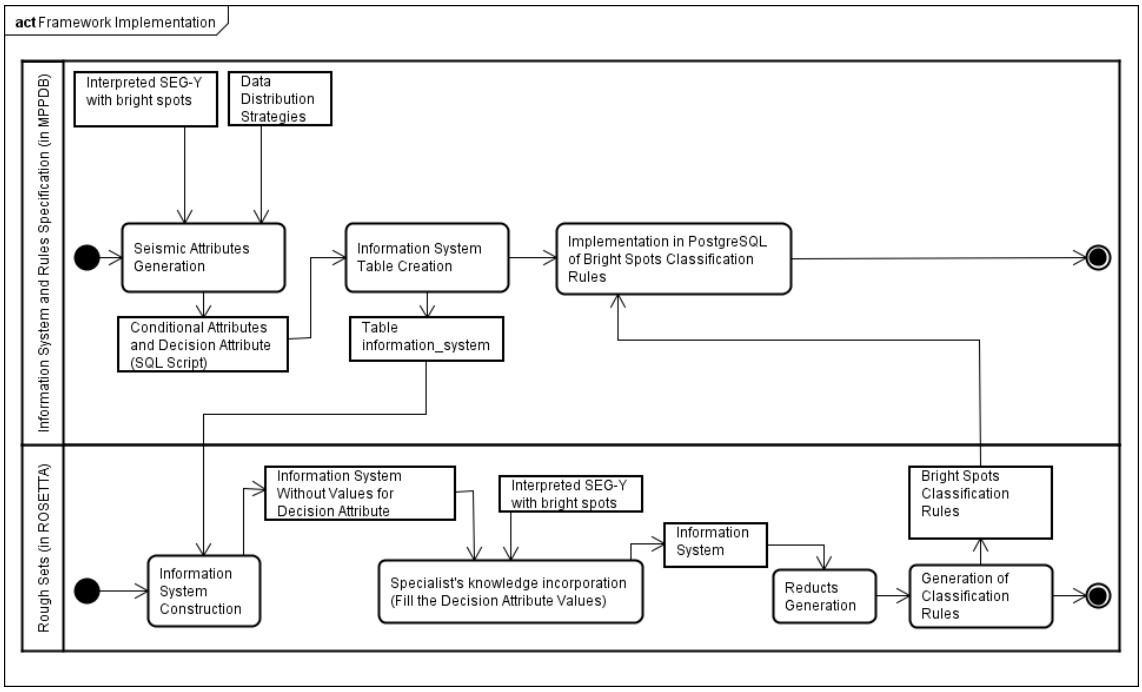


Figure 11. Activity diagram of our framework implementation

The sub-process *Information System and Classification Rules Specification* is carried on using the Greenplum database. However, the same process applies to another database with minor adaptations.

6.1 SEISMIC ATTRIBUTES GENERATION

The first activity of this subprocess is the *Seismic Attributes Generation*. The objective of this activity is to build the information system attributes (see Section 5.1). The *Seismic Attributes Generation* receives the two input artefacts below:

- **Interpreted SEG-Y with bright spots**

  This artefact corresponds to the table *segy* as depicted in Table 6. We consider the table *segy* already created and populated in our database, storing seismic traces from several SEG-Y files. This table has the following columns: (i) *seg_id* to identify the SEG-Y file, (ii) *trace_id* to identify the seismic trace within its respective SEG-Y file, and (iii) *trace_data* that is an array containing the set of amplitudes for the corresponding seismic trace in the same table's row.

Table 6. Table *segy* with seismic trace data from several SEG-Y files

segy_id integer	trace_id integer	trace_data integer[]
1	3	{0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
1	5	{0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,-750,-1295,-15
1	1	{0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,-852,-1490,-12
1	7	{0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,-660,-1591,-14

- **Data distribution strategies**

  Since our framework will work in both ways serial or parallel processing, thus this artefact is a PostgreSQL Script (POSTGRESQL 2015) to determine the data distribution policies if we desire to store and distribute the seismic data within an MPPDB. These policies allocate the data avoiding common problems such as processing skew and data skew (GOLLAPUDI 2013; PIVOTAL 2015), by setting the appropriate distribution keys for each table in our database schema named *seismic*. Thus, we defined a distribution key composed of two attributes for the table *segy*: *segy_id* and *trace_id*. We chose this key for two reasons: (i) the interpreter will work in only one SEG-Y at a time; and (ii) the attribute *trace_id* is directly related to seismic trace that is the object of our information system. If our framework is not in an MPPDB, the present artefact will not be used.

The output of the *Seismic Attributes Generation* activity is the artefact *Conditional Attributes and Decision Attribute*. This artefact comprises the definition of our seismic attributes (defined in Section 5.1) and a SQL Script (stored procedure called *implement_rough_set_information_system*) to generates and populates the temporary table *temp_data* (Table 7) that is required to achieve our conditional attributes in the *information_system* table (Table 8).

Table 7. Table *temp_data* populated with intermediate calculations

trace_id integer	sample_max_index integer	bright_spot_piece character(1)
1	420	t
2	419	t
3	419	t
4	423	t

The table *temp_data* presents the following columns: (i) *trace_id* to identify the seismic trace, (ii) *sample_max_index* to indicate the position of the higher amplitude sample within the respective seismic trace, (iii) *bright_spot_piece* describes if the seismic trace is a piece of bright spots. Such attribute presents the value *t* when it is true that the seismic trace is a piece of bright spots (presence of phase inversion around *sample_max_index*) and, in the opposite case, the value *f* is designated.

6.2 INFORMATION SYSTEM TABLE CREATION

The next activity is the *Information System Table Creation*. *The purpose of this activity is to create and populate the conditional attributes in the table* information_system *(Table 8)* of our database.

Table 8. Table *information_system* populated with the conditional attributes

trace_id integer	bright_spot_pieces_region character(22)	sample_max_indexes_region character(22)	bright_spot boolean
1	>90%TraceRadius	Inclination>4Samples	
9	>90%TraceRadius	Inclination>4Samples	
17	>90%TraceRadius	Inclination>4Samples	
25	>90%TraceRadius	Inclination>4Samples	

The *Information System Table Creation* uses the input artefact *Conditional Attributes and Decision Attribute* by running the *implement_rough_set_information_system* in Greenplum database, as the following query and input parameters below:

```
SELECT seismic.implement_rough_set_information_system (3,16,4,8,'+')
```

- **segy_id**

  The first parameter is *segy_id*; it selects the seismic dataset that corresponds to the seismic section of the F3 volume (inline 250, crossline interval 300 - 1250, time slice interval 0 - 1848). A specialist geologist helped us to choose this section because it was one of the sections with real bright spots samples. The passed parameter value is *3* because the tag for such SEG-Y seismic section is *3*, in the table *segy*.

- **time_search_interval**

  The second parameter is *time_search_interval.* In our framework implementation, a specialist geologist helped us to choose empirically such parameter value equals *16*, which allows covering the vertical size of the major bright spots occurring in our seismic section.

- **sample_time**

  The third parameter is *sample_time*; it informs the sampling time for the samples that constitutes the seismic trace in the SEG-Y. In our framework implementation, the sample time for the chosen SEG-Y is four milliseconds.

- **trace_search_radius**

  The fourth parameter is *trace_search_radius*. In our framework implementation, a specialist geologist helped us to choose empirically such parameter value equals *8*, which yields a neighborhood fitting within the horizontal sizes of the bright spots in our seismic section.

- **phase_reference**

  The fifth parameter is *phase_reference*; it informs that the referential phase is positive for the chosen SEG-Y in our framework implementation.

6.3 INFORMATION SYSTEM CONSTRUCTION

After populating the conditional attributes in the table *information_system*, we go to the first activity *Information System Construction* of the sub-process *Rough Sets.* The aim of *Information System Construction* is to import to Rosetta toolkit the registers from the table *information_system* from our database. This activity produces the output

artefact *Information System without Values for Decision Attribute* that consists of our decision table (see section 2) in Rosetta toolkit.

6.4 SPECIALIST'S KNOWLEDGE INCORPORATION

In the present activity called *Specialist's knowledge incorporation,* the objective is to fill the decision attribute values in the input artefact *Information System without Values for Decision Attribute*. Such values are based on the input artefact *Interpreted SEG-Y with Bright Spots* that is one of the F3 seismic sections containing bright spots (see Section 5.1). In our implementation, we choose the seismic section in the inline 250, but such a seismic section could be any other since it contained samples of bright spots (Figure 12). Therefore the specialist's knowledge comes from the interpreters who worked on such F3 seismic section.

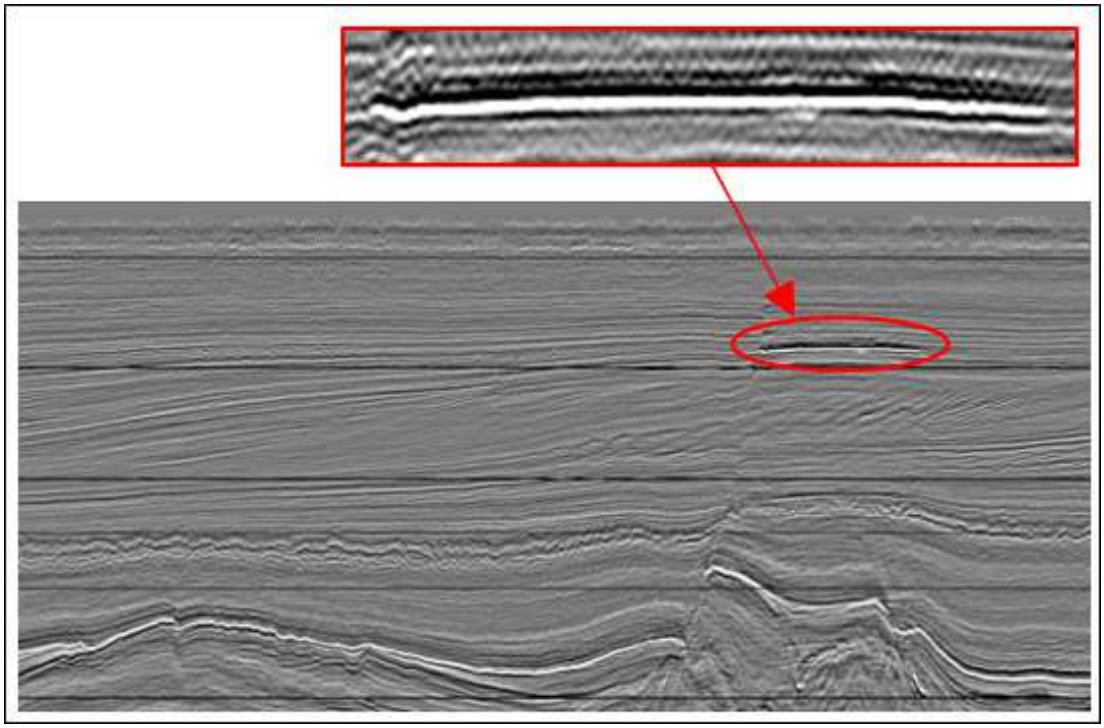
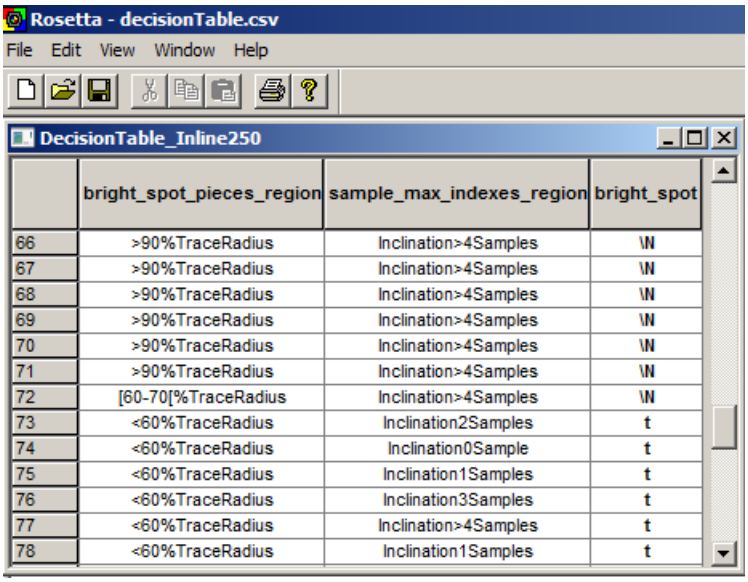


Figure 12. On the top, a magnification of bright spots region. On the bottom, the interpreted seismic section (inline 250) of F3 volume

The assignment of the decision attribute values is done manually by the developer of this framework, editing directly in the decision table *Information System without Values for Decision Attribute*, inputting value *t* for the seismic trace objects that were interpreted as bright spots in F3 seismic section (inline 250). Thus, the activity *Specialist's knowledge incorporation* produces the output artefact *Information System*,

which consists of a decision table in Rosetta toolkit (Figure 13) containing the values of conditional attributes for all the objects of the information system, and the values of the decision attribute for the objects that states a bright spots occurrence.



	bright_spot_pieces_region	sample_max_indexes_region	bright_spot
66	>90%TraceRadius	Inclination>4Samples	\N
67	>90%TraceRadius	Inclination>4Samples	\N
68	>90%TraceRadius	Inclination>4Samples	\N
69	>90%TraceRadius	Inclination>4Samples	\N
70	>90%TraceRadius	Inclination>4Samples	\N
71	>90%TraceRadius	Inclination>4Samples	\N
72	[60-70[%TraceRadius	Inclination>4Samples	\N
73	<60%TraceRadius	Inclination2Samples	t
74	<60%TraceRadius	Inclination0Sample	t
75	<60%TraceRadius	Inclination1Samples	t
76	<60%TraceRadius	Inclination3Samples	t
77	<60%TraceRadius	Inclination>4Samples	t
78	<60%TraceRadius	Inclination1Samples	t

Figure 13. Information system as decision table in Rosetta toolkit

6.5 REDUCTS GENERATION AND CLASSIFICATION RULES GENERATION

The next activities are *Reducts Generation* and *Classification Rules Generation*. This activity receives as input the artefact *Information System* and is responsible for running a supervised training process in Rosetta (see Section 5.2) generating an information system *reduct* if applicable and classification rules to detect bright spots. The output artefact *Bright Spots Classification Rules* (Table 9) is provided by Rosetta toolkit and comprises: (i) rules for the information system *reduct*; such rules present in their expression only one conditional attribute; (ii) rules for the information system that contain the two conditional attributes in their conjunctive expression. Once finish this activity, the framework accomplishes the subprocess *Rough Sets*.

Table 9. Table with classification rules in Rosetta toolkit

	Rule	Accuracy
1	bright_spot_pieces_region(<60%TraceRadius) AND sample_max_indexes_region(Inclination3Samples) => bright_spot(\N) OR bright_spot(t)	0.5, 0.5
2	bright_spot_pieces_region(>90%TraceRadius) AND sample_max_indexes_region(Inclination>4Samples) => bright_spot(\N) OR bright_spot(t)	0.98, 0.02
3	bright_spot_pieces_region(>90%TraceRadius) AND sample_max_indexes_region(Inclination3Samples) => bright_spot(\N)	1.0
4	bright_spot_pieces_region([70-80[%TraceRadius) AND sample_max_indexes_region(Inclination0Sample) => bright_spot(\N)	1.0
5	bright_spot_pieces_region(>90%TraceRadius) AND sample_max_indexes_region(Inclination1Samples) => bright_spot(\N) OR bright_spot(t)	0.92, 0.08
6	bright_spot_pieces_region(>90%TraceRadius) AND sample_max_indexes_region(Inclination2Samples) => bright_spot(\N)	1.0
7	bright_spot_pieces_region(>90%TraceRadius) AND sample_max_indexes_region(Inclination0Sample) => bright_spot(t) OR bright_spot(\N)	0.72, 0.28
8	bright_spot_pieces_region(<60%TraceRadius) AND sample_max_indexes_region(Inclination>4Samples) => bright_spot(\N) OR bright_spot(t)	0.25, 0.75
9	bright_spot_pieces_region([70-80[%TraceRadius) AND sample_max_indexes_region(Inclination>4Samples) => bright_spot(\N)	1.0
10	bright_spot_pieces_region(<60%TraceRadius) AND sample_max_indexes_region(Inclination2Samples) => bright_spot(t)	1.0
11	bright_spot_pieces_region(<60%TraceRadius) AND sample_max_indexes_region(Inclination0Sample) => bright_spot(t)	1.0
12	bright_spot_pieces_region(<60%TraceRadius) AND sample_max_indexes_region(Inclination1Samples) => bright_spot(t)	1.0
13	bright_spot_pieces_region([70-80[%TraceRadius) AND sample_max_indexes_region(Inclination1Samples) => bright_spot(t)	1.0
14	bright_spot_pieces_region([80-90[%TraceRadius) => bright_spot(\N)	1.0
15	bright_spot_pieces_region([60-70[%TraceRadius) => bright_spot(\N)	1.0

In Table 9, for each rule, we have an associated value for accuracy, which corresponds respectively to decisions outputted by the rule. For example, the seventh rule in Table 9 has accuracy equals 0.72 when identifying not bright spots and has accuracy equals 0.28 when detecting true bright spots.

6.6 IMPLEMENTATION IN POSTGRESQL OF BRIGHT SPOTS CLASSIFICATION RULES

The following activity is the *Implementation in PostgreSQL of Bright Spots Classification Rules***.** This activity receives as input the artefact *Bright Spots Classification Rules,* and this activity is in charge of selecting and implementing part of the classification rules as PostgreSQL rules (POSTGRESQL 2015) in our database. Since some rules within *Bright Spots Classification Rules* do not indicate bright spots or have a low accuracy to indicate bright spots, we applied the following selection heuristics. First, all the deterministic rules (see Section 2.3.3) that are the ones with accuracy value equals 1.0, in Table 9 they are the *information system rules* 10, 11, 12, 13 for bright spots presence and, the *information system reduct rules* 14 and 15 for bright spots absence. Next, among the non-deterministic rules (see Section 2.3.3), we selected those with more than 70% of accuracy for bright spots presence, in Table 9, they are the *information system rules* 7 and 8. An expert geologist helped us empirically choose this percentage value, since above this value of accuracy, significant detection of bright spots was still possible. Finally, the selected rules were manually codified as PostgreSQL rules by the developer of this framework, finishing the framework implementation.

7 SEISMIC INTERPRETATION FRAMEWORK

In our framework, we have a computational process to identify a set of bright spots (*BS*), and the user of our framework inputs query parameters (*QP*) specifying the options for *BS* detection. Thus, based on *QP* the process populates the values for *C* in *IS*; then the computational process proceeds to detect *BS* in *IS*, assigning the value *True* for *D* of objects in *IS* that fits the case of a *BS*. The computational process is provided with a set of classification rules to detect *BS*. Such rules were generated based on Rough Set Theory as described in Section 2.3 and Section 6.

7.1 FRAMEWORK ARCHITECTURE

The proposed framework has a logical architecture comprising three main modules: API, Processing and Data Storage as depicted in Figure 14. Following we describe the components of each one of these modules.

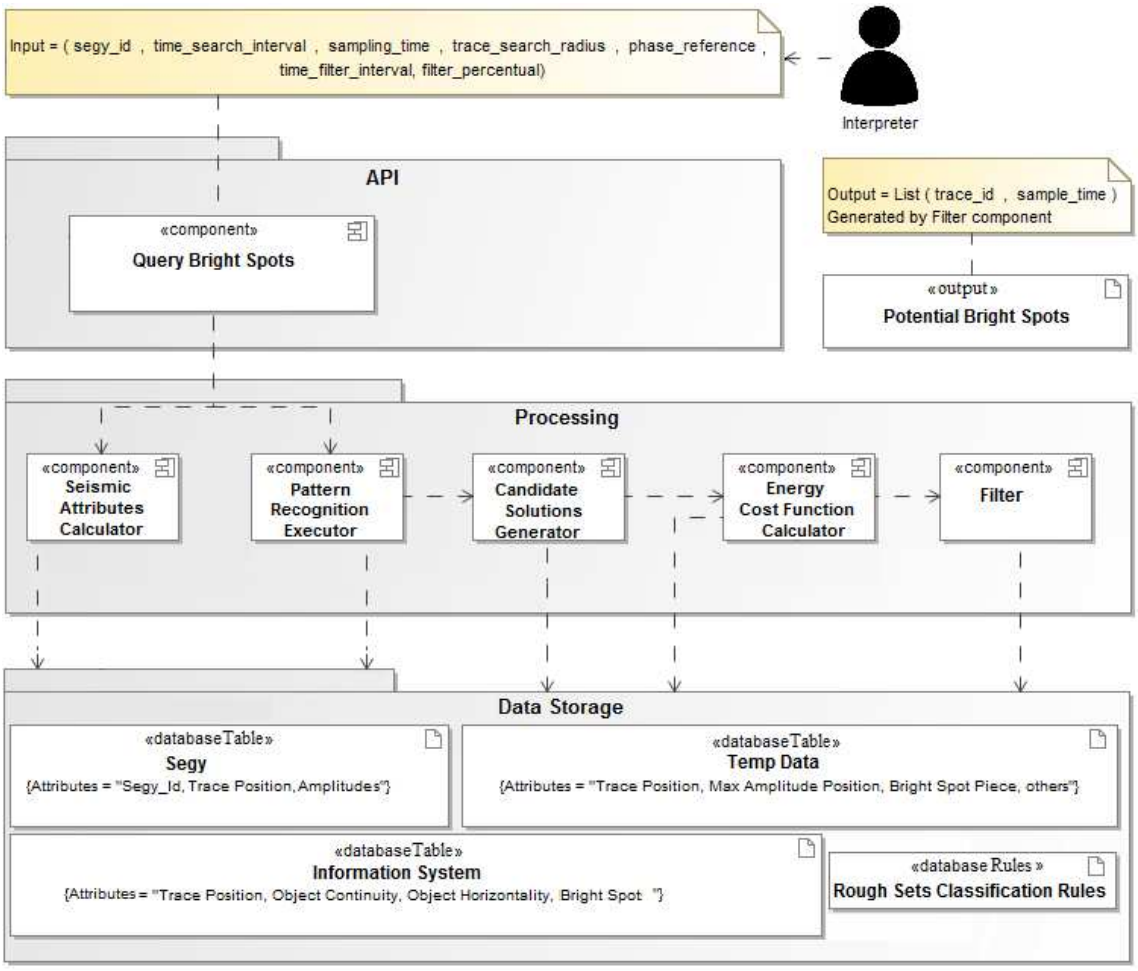


Figure 14. Logical architecture of our seismic interpretation framework

Users access the framework through the *API* module using the *Query Bright Spots* component. Such component is implemented by a stored procedure, which defines a pattern recognition query and has the following parameters (similar to the parameters in Section 6): (i) *segy_id* – specifies the SEG-Y file as seismic dataset to be interpreted; (ii) *time_search_interval* (See Section 5.1); (iii) *sampling_time* – informs the sampling time for the SEG-Y file to be interpreted; (iv) *trace_search_radius* (See Section 5.1); (v) *phase_reference* – value "+" for SEG-Y file with positive amplitude phase reference or value "-" in the opposite case; (vi) *trace_segment_energy* – indicates the number of samples in the subtrace to be considered when calculating the energy, such subtrace starts in the trace sample with higher amplitude value, a null value will indicates to use all the samples within the trace; (vii) *top_percent_filter* – specifies the percentage of best candidates to be kept in the final results; (viii) *rules* – specifies which classification rules must be used, *informationSystem* for information system rules or *reduct* for information system reduct rules.

Regarding the *Processing* module, this is responsible for extracting potential bright spots of the seismic dataset and encompasses the components that are SQL scripts implemented as stored procedures in our database. When the user runs the *Query Bright Spots* such functions will be called in the same order of the *Processing* components below:

- **Seismic Attributes Calculator**

  The *Seismic Attributes Calculator* is responsible for calculating the values of seismic conditional attributes for each object in our information system. Such calculation is performed according to the *Query Bright Spots* parameters.

- **Pattern Recognition Executor**

  The *Pattern Recognition Executor* is in charge of carrying out the pattern recognition, analyzing for each object in the information system the values of seismic conditional attributes. Such values will determine if there are bright spots in the current object, in the affirmative case, the object will receive the value *True* in its decision attribute. The component's execution is according to the component *Query Bright Spots*, in the module *API* and according to the artefact *Rough Sets Classification Rules*, in the module *Data Storage*.

- **Candidates Solutions Generation**

  The objective of *Candidates Solutions Generation* is to select, in the *Information System*, the objects that present the value *True* for the decision attribute. Thus, this component generates as output a set of candidates objects for the pattern recognition result.

- **Energy Cost Function Calculator**

  The responsibility of *Energy Cost Function Calculator* is to apply a cost function for each seismic trace object selected by the component *Candidates Solutions Generation,* calculating the Energy of such objects. The result of *Energy Cost Function Calculator* permits to indicate the objects that present more potential for pore fluid, therefore, objects with higher probability to be bright spots associated with hydrocarbon accumulation.

- **Filter**

  The *Filter* is in charge to mitigate the occurrence of false positives in our result. For this purpose the component ranks decreasingly by the Energy value the candidates from *Candidates Solutions Generation,* filtering to be in the results only the *x*% first top candidates, where *x*% is informed by the parameter *filter* in the component *Query Bright Spots*.

Concerning the *Data Storage* module, the responsibility of this module is to store the seismic dataset, to provide all the necessary tables and rules used by our computational process of our framework, encompassing the following artefacts:

- **Segy**

  The table *Segy* (see Table 6) allocates the SEG-Y files. In the module *Processing*, the component *Seismic Attributes Calculator* accesses *Segy*.

- **Temp Data**

  The table *Temp Data* (see Table 7) stores the intermediate calculations results for the components in the module *Processing*.

- **Information System**

  The table *Information System* (see Table 8) represents our information system. In the module *Processing*, the component *Seismic Attributes Calculator* generates

the values to populate the conditional attributes in the *Information System.* Also, all the others components in *Processing* access this artefact.

- **Rough Sets Classification Rules**

  The *Rough Sets Classification Rules* is a set of PostgreSQL rules (POSTGRESQL 2015) in our database and, such rules provide the *if-then* rules generated at our framework implementation.

The final result of our computational process is the output *Potential Bright Spots*, which consists of a CSV text file, containing a list of coordinates for seismic traces objects that present bright spots potentially associated with hydrocarbon accumulation. The coordinate comprises: (i) trace identification (see *trace_id* in Table 7) and, (ii) sample time of *sample_max_index* (see Table 7).

7.2 COMPUTATIONAL PROCESS DESCRIPTION

In this section, we describe the computational *Seismic Interpretation Support Process* of our Framework to identify bright spots in seismic datasets. The entire process is split into two sub-processes: (i) *Pattern Recognition Query* and (ii) *Information System Customisation*. We envision one user profile in our framework: the *Interpreter*, who handles the query parameters description, according to the bright spots of his/her interest. The UML activity diagram depicted in Figure 15 shows the process as a whole with its sub-processes, input and output artefacts. Following, we describe this activity diagram relating to the logical architecture of our *seismic interpretation framework.*

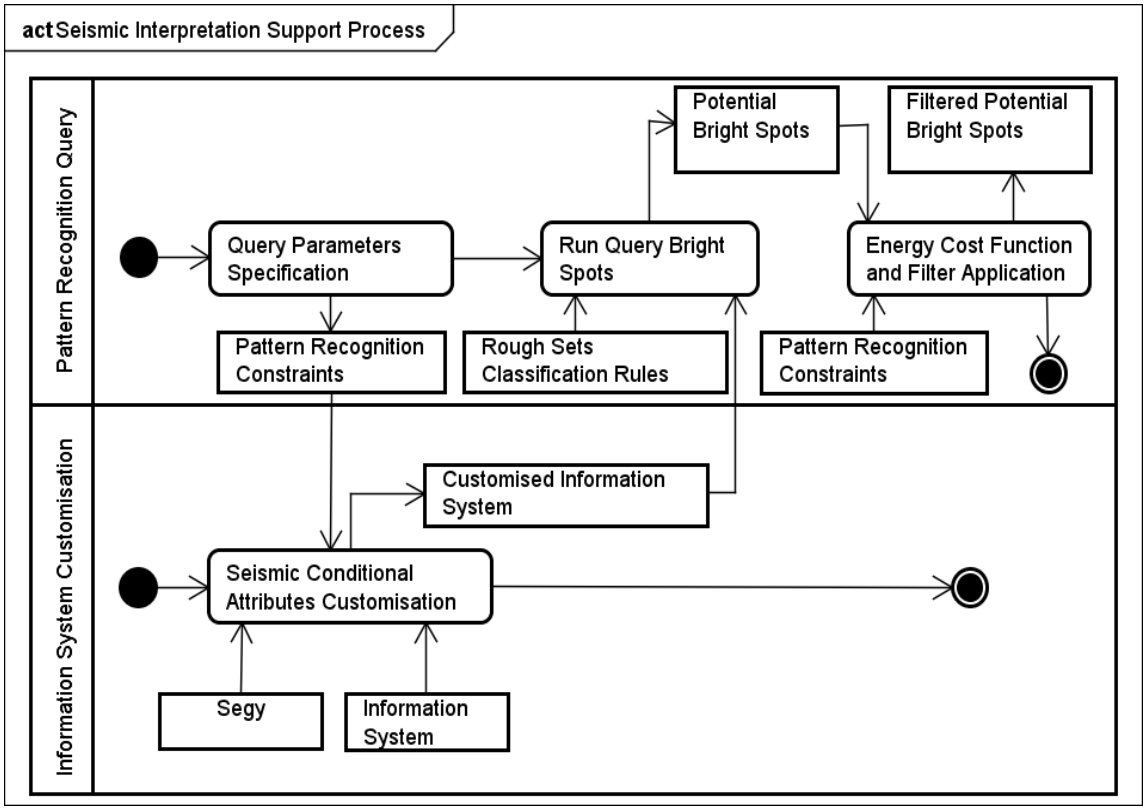


Figure 15. Activity diagram of the computational seismic interpretation support process

- **Query Parameters Specification**

  This activity is the first step of the sub-process *Pattern Recognition Query* and, consists of the specification of the query parameters. *Query Parameters Specification* uses the component *Query Bright Spots* (Figure 14) and generates the *Pattern Recognition Constraints* artefact, containing the parameters inputted by the user (Figure 14).

- **Seismic Conditional Attributes Customisation**

  The purpose of this activity is to customise the seismic conditional attributes, calculating their values according to the *Pattern Recognition Constraints* and, according to others two input artefacts *Segy* and *Information System* (Figure 14 and Figure 15). Also, this activity uses the component *Seismic Attributes Calculator* (Figure 14). Lastly, the present activity produces the artefact *Customised Information System*, which consists of the *Information System* (Figure 14) populated with the values of the seismic conditional attributes for each object.

Once finish this activity, we accomplished the subprocess *Information System Customisation*.

- **Run Query Bright Spots**

After creating the *Customised Information System*, the framework proceeds with the sub-process *Pattern Recognition Query*, where the activity *Run Query Bright Spots* uses the components *Pattern Recognition Executor* and *Candidate Solutions Generator* (Figure 14) to identify the bright spots in the input artefact *Customised Information System*. In addition, this activity has another input artefact the *Rough Sets Classification Rules* (Figure 14 and Figure 15). This activity creates the artefact *Potential Bright Spots*, which corresponds to the *Information System* (Figure 14) with the values of the conditional and decision attributes for objects were classified as bright spots associated with potential hydrocarbon accumulation.

- **Energy Cost Function and Filter Application**

The aim of this activity is to refine the result of the previous activity, using the components *Energy Cost Function Calculator* and *Filter* (Figure 14). This activity receives the input artefacts *Potential Bright Spots* and *Pattern Recognition Constraints*; then the activity calculates the energy for each object and, filters the top percentage with higher energy values, producing the artefact *Filtered Potential Bright Spots* that is analog to our framework output (Figure 14). Finally, we achieve the sub-process *Pattern Recognition Query*, concluding the S*eismic Interpretation Support Process*.

8 EVALUATION

The proposed framework was evaluated through a set of experiments. The first set intends to assess the framework accuracy to detect bright spots as potential hydrocarbon indicators. The second set of experiments evaluates the framework performance in different scenarios. All the evaluations were performed in the MPPDB Greenplum (GOLLAPUDI 2013; PIVOTAL 2015).

8.1 ACCURACY TESTS

Since we built our framework (as described in Section 6), this is ready to be evaluated and after that used by an interpreter. A preliminary examination consists in confronting our results with F3 seismic volume, verifying how close are our detection of bright spots from the real bright spots. Performing this test, we can verify if our framework attends to the functional requisite.

We present four evaluation scenarios to verify the correctness of our framework's results. Such scenarios state examples of the use of our framework, envisaging to cover the cases when executing the framework with no energy filter and with energy filter, then checking how well the filter removes false positives. Also, the scenarios cover the cases when considering the *information system* and when considering the *information system reduct*. Thus we can evaluate the impact of the reduction of our seismic conditional attributes.

Scenario 1: Framework over the information system without energy filter

In Greenplum database, we execute the following query to yield the scenario 1:

```
SELECT seismic.query_bright_spots (1,16,4,8,'+',  , 100, 'informationSystem');
```

Note that the parameter *segy_id* has value *1*, which is related to a SEG-Y that corresponds to a seismic section of F3 volume (inline 200, crossline interval 300 - 1250, time slice interval 0 - 1848). Such seismic section is different from that one used in our framework implementation.

Figure 16 shows the results of our first test over the *information system*. To facilitated the visualization of seismic events, we adopted the representation in grayscale for our seismic section. In the grayscale representation, for a positive phase reference, the higher amplitude value will be black, and the smaller amplitude value will be white, or the opposite if adopted a negative phase reference; the intermediate amplitudes values will present a proportional gray color intensity. Moreover, we put a yellow marker (SCHINELLI 2011) in the samples that are the higher amplitude value of a seismic trace related to the bright spots.

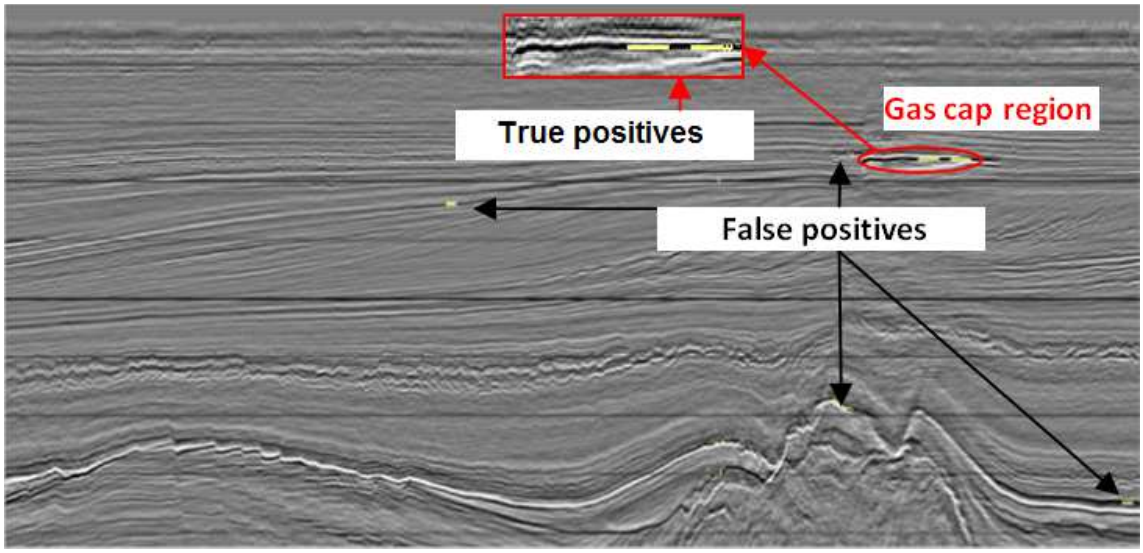


Figure 16. Framework results considering the information system and without energy filter in a seismic section of F3 volume (inline 200)

In scenario 1, our framework detected correctly 32 (*True Positives – TP*) of the 89 bright spots (*Positives – P*). Also, our framework classified correctly 756 (*True Negatives – TN*) of 862 not bright spots (*Negatives – N*). Thus we reached a rate of correct results equals 82.86% and sensitivity (*TP* rate) of 35.95%.

Scenario 2: Framework over the information system with energy filter

In Greenplum database, we execute the query below to achieve the scenario 2. Figure 17 shows the results of the present scenario evaluation.

```
SELECT seismic.query_bright_spots (1,16,4,8,'+',  ,80, 'informationSystem');
```

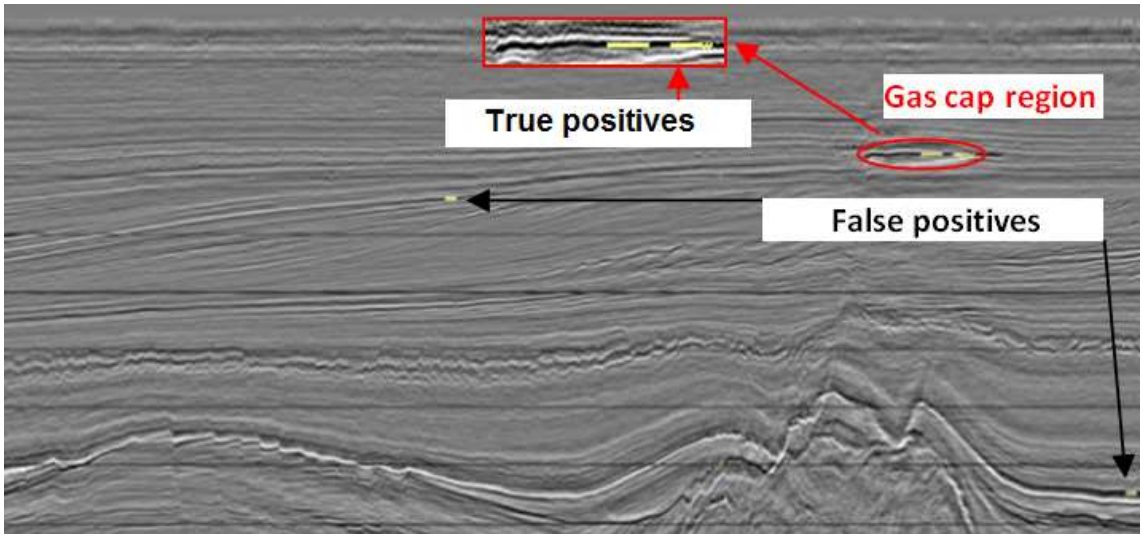


Figure 17. Framework results considering the information system and with energy filter in a seismic section of F3 volume (inline 200)

In scenario 2, our framework detected correctly 32 (*TP*) of the 89 bright spots (*P*). Also, our framework classified correctly 775 (*TN*) of 862 not bright spots (*N*). Thus we reached a rate of correct results equals 84.85% and sensitivity of 35.95%. Comparing with the scenario 1, the scenario 2 kept the correct results and minimized the *False Positives* (*FP*), validating the filter purpose.

Scenario 3: Framework over the information system *reduct* without energy filter

To produce the scenario 3, in Greenplum database we execute the query below. Figure 18 depicts the results of this present evaluation scenario.

```
SELECT seismic.query_bright_spots (1,16,4,8,'+',  ,100, 'reduct');
```

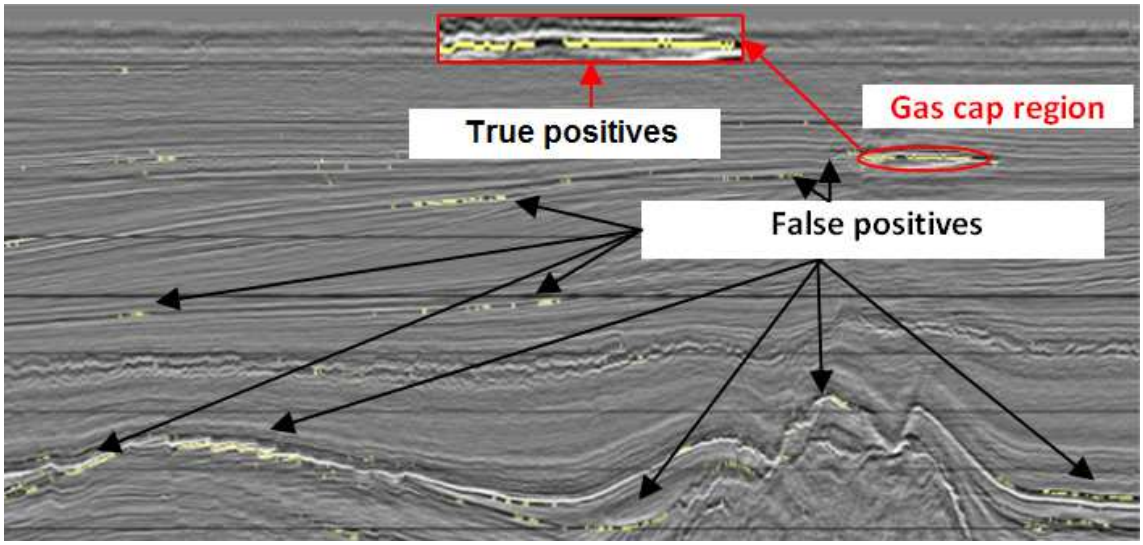


Figure 18. Framework results considering the information system *reduct* and without energy filter in a seismic section of F3 volume (inline 200)

In scenario 3, our framework detected correctly 79 (*TP*) of the 89 bright spots (*P*). Also, our framework classified correctly 129 (*TN*) of the 862 (*N*) bright spots. Thus we reached a rate of correct results equals 21.87% and sensitivity of 88.76%. For such *reduct*, the high incidence of *FP* is because the classification rules address only one seismic attribute, what makes the rules less restrictive to classify an object as bright spots.

Scenario 4: Framework over the information system *reduct* with energy filter

To achieve the scenario 4, in Greenplum database we execute the query below. Figure 19 shows the results for Scenario 4.

```
SELECT seismic.query_bright_spots (1,16,4,8,'+', 16, 25, 'reduct');
```

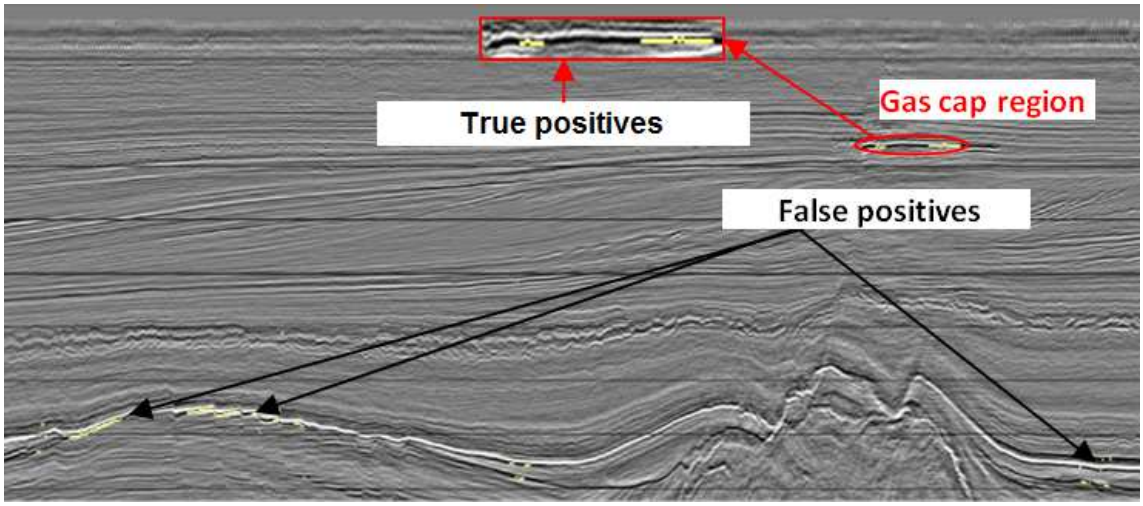


Figure 19. Framework results considering the information system reduct and with energy filter in a seismic section of F3 volume (inline 200)

In scenario 4, our framework detected correctly 36 (*TP*) of the 89 bright spots (*P*). Also, our framework classified correctly 647 (*TN*) of the 862 (*N*) bright spots. Thus we reached a rate of correct results equals 71.81% and sensitivity of 40.44%. Consequently, the filter minimized the number of *FP*, but also eliminate some correct results compared to the scenario 3 results.

In the previous outcomes of the proposed framework, we can see in the Figures 16, 17, 18 and 19 the oval shape indication (local magnification in the rectangular shape indication) showing the region with bright spots (black track) associated with a gas accumulation. Some pieces of the bright spots were properly detected by our framework as indicated by the yellow marker over the black track. Also, for example in Figure 19 at the bottom left corner, our framework wrongly highlight (yellow marker) some horizons as bright spots. Therefore our framework may present some false positives in its results.

On another perspective, even though our framework does not find all the bright spots, all the undetected bright spots (*False Negatives – FN*) are noticeably close to the majority of the *TP*, thus our framework leads the interpreter to focus just on relevant seismic sections and section regions with more probability of bright spots occurrence, saving him/her from analyzing all the sections and analyzing all data in the seismic section. Therefore our framework can make the seismic interpretation activity faster.

According to the Section 3, there is in the literature another semi-automatic method to identify bright spots (FARFOUR et al. 2012). However, we could not perform an evaluation to compare the accuracy of the results from our proposed solution and the results from that other approach, since the results presentations are not clear in such related work and its approach is not available, then we could not reproduce such results to proceed with the comparison.

8.2 PERFORMANCE TESTS

To evaluate how well the proposed framework performs in serial and parallel environments, we deployed the Greenplum MPPDB on a cluster and changed its configuration from 1 to 12 Greenplum computing segments. Table 10 presents the configuration of each cluster node.

Table 10. Cluster node configuration

Cores	4
Memory	30 GB
Hard Disk	400 GB
Operating System	CentOS 6 x64

The configuration of cluster nodes was based on the Greenplum's manual (PIVOTAL 2015) to create a robust cluster to perform the tests. The Greenplum master server and the computing segments follow the number of cores in a cluster node, thus as the nodes have four cores each one, it is possible to configure up to four segments for each node. About the master, it was allocated in one separated cluster node.

We analyzed two groups of configuration: serial configuration, with one segment; and parallel configuration, with 2, 4, 8, and 12 segments. Furthermore, the tests were performed on seismic datasets containing 10, 20, 40, 80 and 120 times the seismic dataset from the F3 volume, which has 50 MB covering only the amplitudes of the seismic traces. Therefore, we totalize 25 scenarios, covering the combinations among the numbers of computing segments and the multiplication factors of the seismic dataset:

(1) 1 computing segment and 10 times the dataset;
(2) 1 computing segment and 20 times the dataset;
(3) 1 computing segment and 40 times the dataset;

(4) 1 computing segment and 80 times the dataset;
(5) 1 computing segment and 120 times the dataset;
(6) 2 computing segments and 10 times the dataset;
(7) 2 computing segments and 20 times the dataset;
(8) 2 computing segments and 40 times the dataset;
(9) 2 computing segments and 80 times the dataset;
(10) 2 computing segments and 120 times the dataset;
(11) 4 computing segments and 10 times the dataset;
(12) 4 computing segments and 20 times the dataset;
(13) 4 computing segments and 40 times the dataset;
(14) 4 computing segments and 80 times the dataset;
(15) 4 computing segments and 120 times the dataset;
(16) 8 computing segments and 10 times the dataset;
(17) 8 computing segments and 20 times the dataset;
(18) 8 computing segments and 40 times the dataset;
(19) 8 computing segments and 80 times the dataset;
(20) 8 computing segments and 120 times the dataset;
(21) 12 computing segments and 10 times the dataset;
(22) 12 computing segments and 20 times the dataset;
(23) 12 computing segments and 40 times the dataset;
(24) 12 computing segments and 80 times the dataset;
(25) 12 computing segments and 120 times the dataset;

8.2.1 Initial experimental results

The initial results are the total processing time of each scenario; Table 11 shows these times in seconds, where the presented values are the processing time average of three executions for each scenario.

Table 11. Processing time varying the number of segments and the multiplication factor of the dataset

Dataset Multiplication factor	Number of Computing Segments: 1	2	4	8	12
10	227,8	115,5	66,3	31,7	21,0
20	464,6	252,2	135,7	60,2	39,3
40	941,4	480,1	253,9	131,4	85,4
80	1989,4	1063,7	515,5	252,9	169,1
120	3564,1	1805,4	901,5	453,3	317,9

The calculated statistics for each scenario cover the values of mean, standard deviation (*SD*) and coefficient of deviation (*CV*), which are depicted in Table 12. Regarding the obtained *CV* values, which are small, we conclude that the standard deviation is minimal; consequently, three runs per scenario are sufficient to obtain a reliable and representative average of the processing time. Furthermore, in Table 13 we can see the confidence interval for each scenario in our experimental results.

Looking at the results in Table 11, we can notice a reduction in processing time as we increase the parallelism. Nevertheless, for a better analysis of performance, according to others studies of distributed/parallel computation (SHAFER et al. 1996; ONG 2010), performance assessment considers the measurement of *speedup*, *scaleup,* and *sizeup*, which are present in the following subsections.

Table 12. Statistics of the initial experimental results

Dataset Multiplication factor	Number of Computing Segments: 1 mean	1 SD	1 CV %	2 mean	2 SD	2 CV %	4 mean	4 SD	4 CV %	8 mean	8 SD	8 CV %	12 mean	12 SD	12 CV %
10	227,8	3,5	1,6	115,5	0,4	0,4	66,3	0,2	0,4	31,7	0,3	0,9	21,0	0,4	1,8
20	464,6	6,1	1,3	252,2	0,3	0,1	135,7	0,6	0,5	60,2	0,7	1,2	39,3	0,4	0,9
40	941,4	3,2	0,3	480,1	0,2	0,0	253,9	0,2	0,1	131,4	0,6	0,4	85,4	0,7	0,8
80	1989,4	5,3	0,3	1063,7	6,6	0,6	515,5	0,5	0,1	252,9	1,9	0,8	169,1	0,2	0,1
120	3564,1	5,5	0,2	1805,4	0,4	0,0	901,5	1,6	0,2	453,3	2,7	0,6	317,9	3,4	1,1

Table 13. Confidence intervals for each scenario

		Number of Computing Segments														
		1			2			4			8			12		
Dataset Multiplication factor	10	227,8	±	4,0	115,5	±	0,5	66,3	±	0,3	31,7	±	0,3	21,0	±	0,4
	20	464,6	±	6,9	252,2	±	0,3	135,7	±	0,7	60,2	±	0,8	39,3	±	0,4
	40	941,4	±	3,6	480,1	±	0,2	253,9	±	0,2	131,4	±	0,7	85,4	±	0,8
	80	1989,4	±	6,0	1063,7	±	7,4	515,5	±	0,6	252,9	±	2,2	169,1	±	0,2
	120	3564,1	±	6,2	1805,4	±	0,4	901,5	±	1,8	453,3	±	3,0	317,9	±	3,9
		confidence level 95%														

8.2.2 Speedup

Speedup assesses the ability of the parallelism to optimize processing time. It is defined as the ratio of the serial processing time to the parallel processing time (ONG 2010). Thus, speedup can be expressed as in Equation 5:

$$\text{Speedup}(s) = \frac{\text{t}(1)}{\text{t}(s)} \qquad (5)$$

Where s is the number of computing segments, t(1) is the processing time of the proposed framework on one computing segment and, t(s) is the processing time on the parallel configuration with s computing segments. An ideal parallelism demonstrates linear speedup, for example, an infrastructure with m times the number of segments produces a speedup of m. Nevertheless, the linear speedup is complicated to accomplish since the communication cost rises with the number of computing segments. Figure 20 shows de speedup evaluation results of our framework in each scenario.

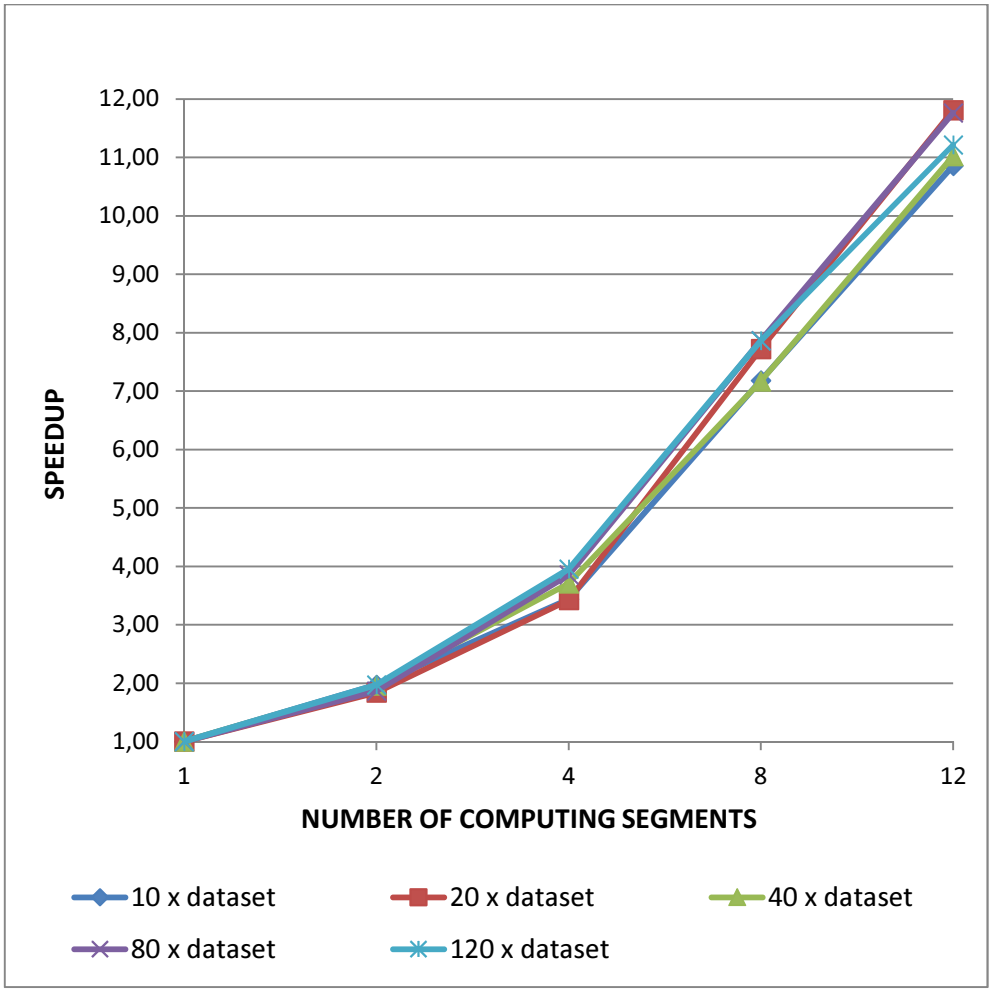


Figure 20. Framework speedup results

From the results acquired, the proposed framework demonstrates satisfactory speedup since the speedup factors progress at about the same rate as we increase the parallelism. In some scenarios the speedup factor is slightly below the proportion of the increased quantity of computing segments, such difference is probably linked to the cluster network cost.

8.2.3 Scaleup

In the scaleup evaluation, we aim to verify the performance of our framework as we increase the number of computing segments for a constant sub-volume of the F3 seismic dataset (SHAFER et al. 1996). Figure 21 shows the results of this set of sensitive experiments.

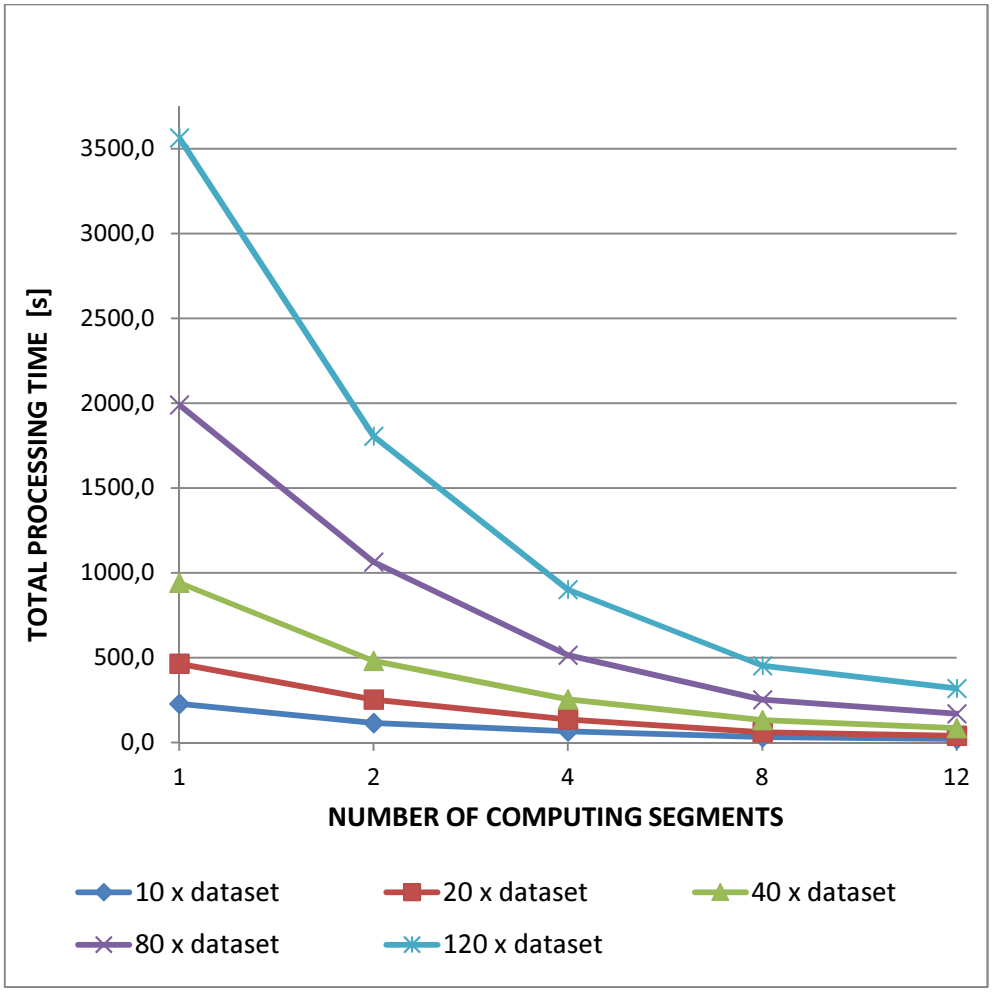


Figure 21. Processing time scaleup results

Also, we want to evaluate the performance of the proposed framework when increasing the number of computing segments and the dataset size proportionally (ONG 2010). In such context we have a scaleup factor that can be expressed as in Equation 6:

$$\text{Scaleup}(s) = \frac{\text{t}(1,D)}{\text{t}(s,sD)} \qquad (6)$$

Where s is the number of computing segments, $\text{t}(1, D)$ is the processing time of the framework on 1 computing segment with dataset size equals D, $\text{t}(s, sD)$ is the processing time with s computing segments and data size equals s times D. An ideal parallelism demonstrates a constant scaleup with growing number of computing segments and dataset size. In Figure 22 we present the calculated scaleup factors for our framework.

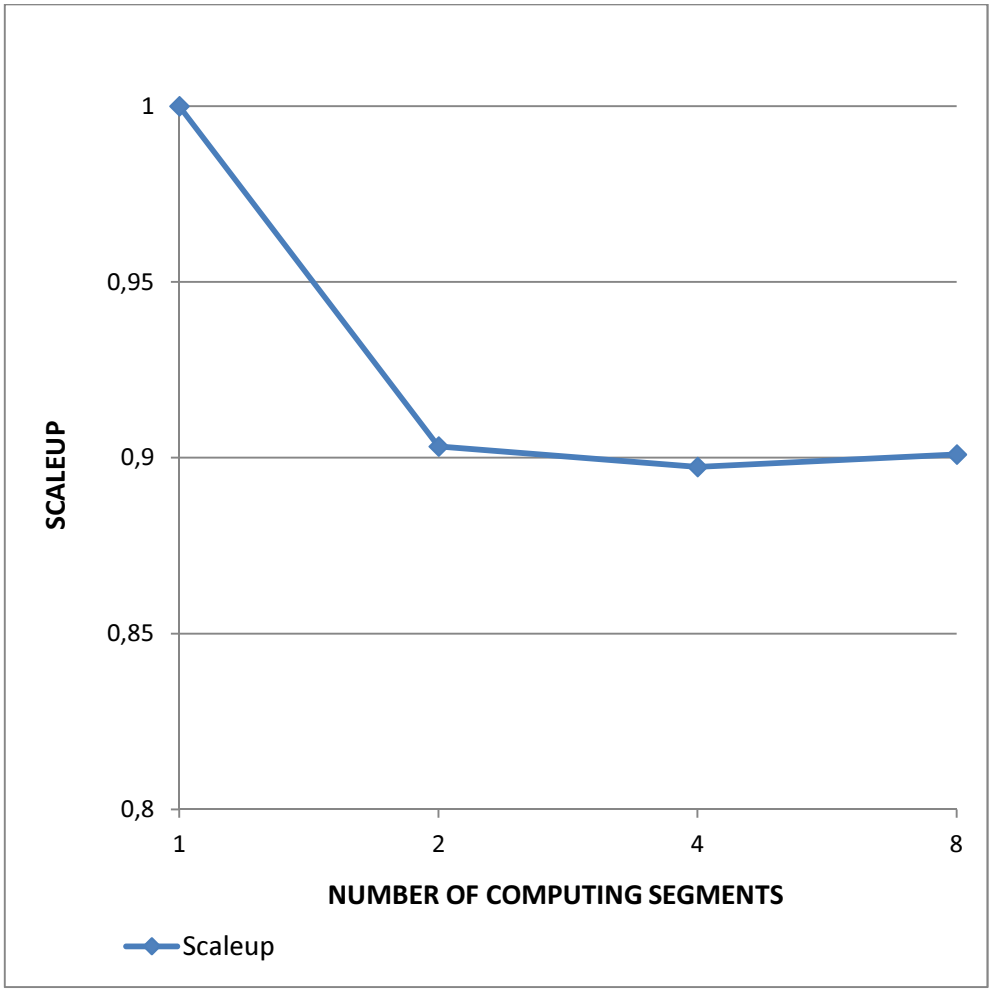


Figure 22. Framework scaleup factor results

Concerning the processing time decreasing in Figure 21 and, the small degradation of the scaleup factor (see Figure 22), these results indicate that the proposed framework is scalable since the performance is improved as we increase the number of computing segments in the Greenplum MPPDB. The decrease in the scaleup factor was probably due to the cluster network cost.

8.2.4 Sizeup

In the sizeup assessment, we examine the performance of our framework as we increase the size of the seismic dataset and the number of segments is fixed (ONG 2010). The sizeup factor can be expressed as in Equation 7:

$$\text{Sizeup}(s,n) = \frac{\text{t}(s,nD)}{\text{t}(s,D)} \qquad (7)$$

Where s is the number of computing segments, and n is the incremental factor of the dataset size. T(s, D) is the framework processing time on s computing segments, and dataset size equals D; T(s, nD) is the processing time with s computing segments, and dataset size equals n times D.

The results of the sizeup tests are presented in Figure 23 and Figure 24.

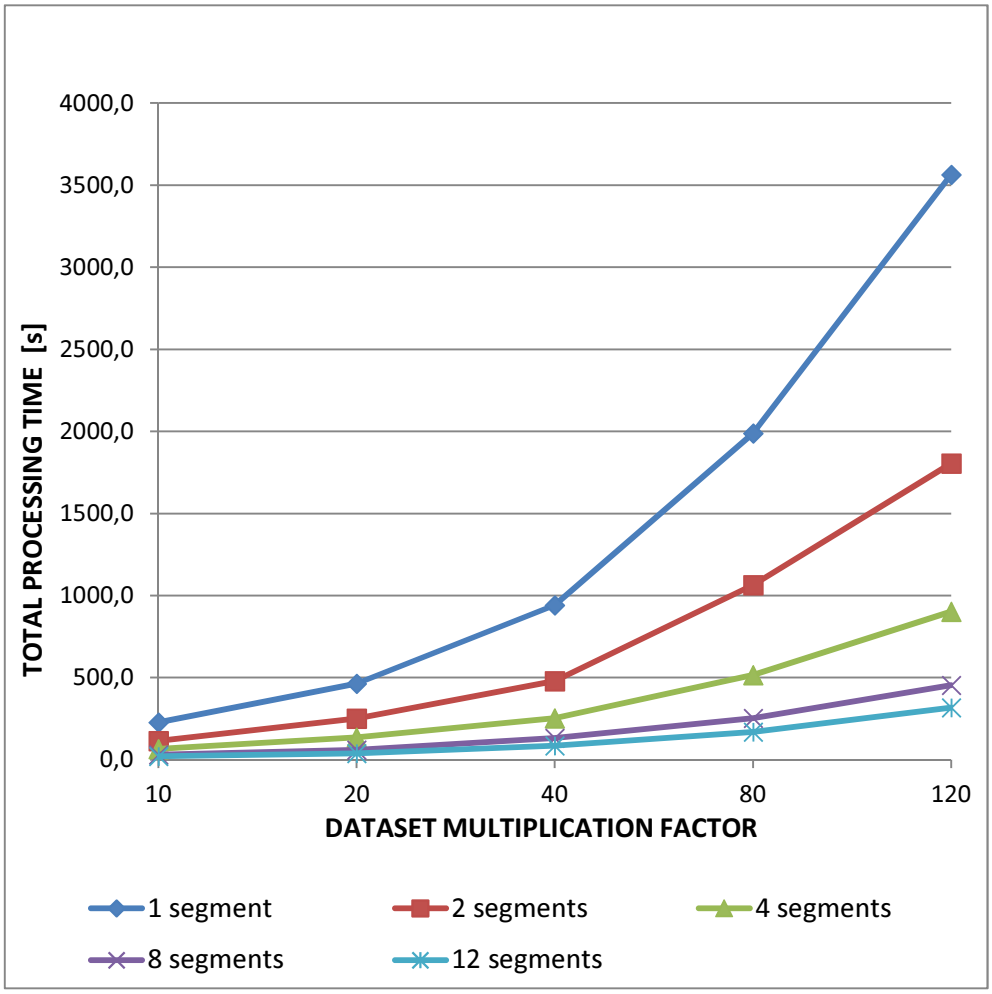


Figure 23. Framework processing time sizeup results

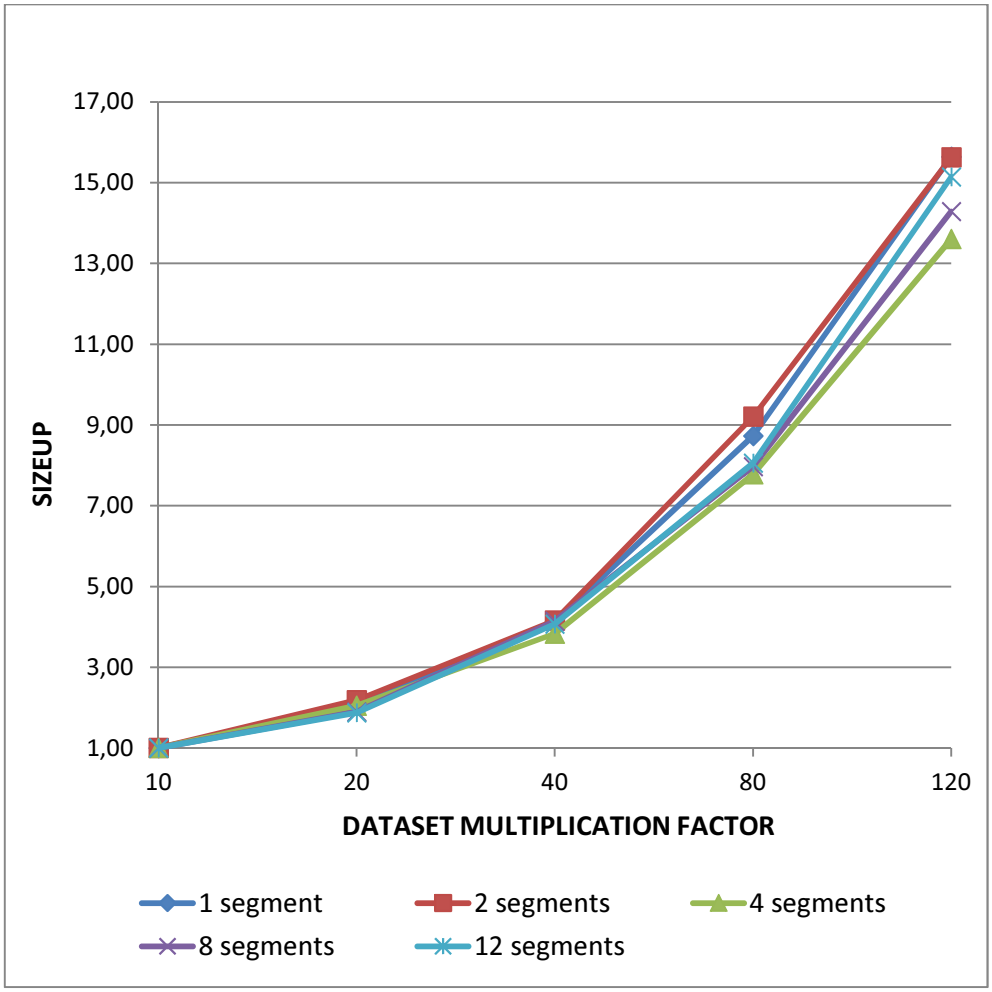


Figure 24. Framework sizeup factor results

We can conclude that the sizeup results of our framework are satisfactory and coherent, as the processing time and sizeup factor increase approximately equal the proportion that the seismic dataset grows, as depicted in Figure 23 and Figure 24. In some scenarios, the sizeup factor increases slightly above the expected probably due to the cluster network cost.

9 CONCLUSION

The research presented here has demonstrated the success of using Rough Set in the proposed seismic interpretation framework, classifying seismic datasets, incorporating specialist's knowledge into the process, and generating classification rules that work for different datasets.

Using a real seismic dataset, which contains bright spots, tests show that our framework can find a suitable pattern of bright spots. By the results of our accuracy tests, we conclude that our framework is a helpful support in the seismic interpretation since the interpreter will focus on the seismic sections and regions where our framework indicates bright spots, saving the interpreter from analyzing and processing all dataset. Therefore the seismic interpretation becomes faster.

Unfortunately, the proposed framework may present some false positives. However, the user can minimize such false positives by tuning the query parameters to detect bright spots. In our framework, the small number of seismic conditional attributes possibly is the reason for the false positives. As a future work, we plan to increase the number of conditional attributes, to achieve more restrictive classification rules and, therefore we expect to reduce the false positives.

Envisioning the interpretation of huge seismic datasets, our framework is prepared to run in an MPPDB if necessary, allocating the data avoiding processing and data skew, therefore ensuring a satisfactory processing time. Concerning the scenario with F3 dataset increased in 120 times, the performance tests show that moving from serial to parallel processing, we reduce the framework processing time in around 49%, 75%, 87% and 91% running in 2, 4, 8 and 12 computing segments respectively. In addition, the tests show that our framework is scalable, the processing times conforms to the sizeup tests results and, the framework presents a satisfactory speedup.

The quality of the results of the proposed framework is in parts determined by the quality of the query parameters, which customise the pattern recognition process for bright spots. Since the interpreter is who inputs such parameters, the framework is relatively dependent on user expertise and experience. Nevertheless, despite such dependency, we believe the framework will be a useful support process for the interpreter in the seismic interpretation activity.

As further developments, we plan to explore the interconnection between rough sets and dependencies in relational databases (KEEN et. al 1994) since one question that dependencies try to answer is what data is relevant to the user's query. In this potential direction we will focus on three types of dependencies: (i) the functional dependencies that are restrictive, assuming the relationship holds for all data under consideration, that there are no exceptions to the rule; (ii) partitioning dependencies that loosen the requirements slightly, they allow one X-value to determine more than one Y-value; and (iii) inductive dependencies are the least restrictive of these dependencies. They allow the data to be loosely grouped, by allowing the groups to overlap in some ways.

REFERENCES

ANDERSON, P. F. Comparing post-stack AVO inversion to pre-stack Inversion for estimating rock properties. In: CSPG/CSEG/CWLS GEOCONVENTION, 2009, Calgary. **Proceedings**... [S.l.: s.n.], 2009.

BAKKE, J. Ø. H.; GRAMSTAD, O.; SØNNELAND, L. **Seismic DNA**: a novel visually guided search method using non-local search and multi-attribute data sets. In: STEEPLES, D. **SEG Technical Program Expanded Abstracts**. [S.l.: s.n.], 2012 . p. 1-5.

BASIR, H. M.; JAVAHERIAN, A.; YARAKI, M. T. Multi-attribute ant-tracking and neural network for fault detection: a case study of an Iranian oilfield. **Journal of Geophysics and Engineering**, Bristol, v. 10, n. 1, 2013.

CHIBURIS, E. et al. Hydrocarbon detection with AVO. **Oilfield Review**, Houston, v. 5, n. 1, p. 42-50, 1993.

CLIFFORD, A.; AMINZADEH, F. Gas detection from absorption attributes and amplitude versus offset with artificial neural networks in Grand Bay Field. In: FOMEL, S. **SEG Technical Program Expanded Abstracts**. [S.l.: s.n.], 2011. p. 375-380.

DGB EARTH SCIENCES. **Netherlands Offshore F3 Block - Complete**. 2015a. Disponível em: <https://www.opendtect.org/osr/pmwiki.php/Main/NetherlandsOffshoreF3BlockComplete4GB>. Acesso em: 13 jan. 2015.

_____. **OpendTect User Documentation**: 6.0. 2015b. Disponível em: < http://doc.opendtect.org/6.0.0/doc/od_userdoc/Default.htm>. Acesso em: 1 dez. 2016.

FARFOUR, M. et al. Seismic attributes combination to enhance detection of bright spots associated with hydrocarbons. **Geosystem Engineering**, [S.l.], v. 15, n. 3, p. 143-150, 2012.

FIGUEIREDO, A. M. **Mapeamento automático de horizontes e falhas em dados sísmicos 3D baseado no algoritmo de gás neural evolutivo**. 2007. 79 p. Dissertação (Mestrado em Informática)-Pontifícia Universidade Católica do Rio de Janeiro, Rio de Janeiro, 2007.

GEORGE, D. **Seismic exploration**: taking another look at bright spots. 1997. Disponível em: <http://www.offshore-mag.com/articles/print/volume-57/issue-3/news/exploration/seismic-exploration-taking-another-look-at-bright-spots.html>. Acesso em: 1 dez. 2015.

GERHARDT, A. L. B. **Aspectos da visualização volumétrica de dados sísmicos**. 1998. 99 p. Dissertação (Mestrado em Engenharia Civil)-Pontifícia Universidade Católica do Rio de Janeiro, Rio de Janeiro, 1998.

GOLLAPUDI, S. **Getting started with Greenplum for big data analytics**. Birmingham: Packt Publishing, 2013.

HAMPSON, D. P.; SCHUELKE, J. S.; QUIREIN, J. A. Use of multi-attributes transforms to predict log properties from seismic data. **Geophysics**, Tulsa, v. 66, n. 1, p. 220-236, 2001.

HERRERA, V. M.; RUSSELL, B.; FLORES, A. Neural networks in reservoir characterization. **The Leading Edge**, Tulsa, v. 25, n. 4, p. 402-411, 2006.

HONGJIE, L. et al. Seismic attribute reduction method and its application. **Information Technology Journal**, v. 13, n. 14, p. 2326-2333, 2014.

JOLLIFFE, I. T. **Principal component analysis**. 2. ed. Berlin: Springer, 2002.

KEEN, D.; RAJASEKAR, A. Rough sets and data dependencies. In: ALAGAR, V. S.; BERGLER, S.; DONG, F. Q. (Ed.). **Incompleteness and Uncertainty in Information Systems**. London: Springer, 1994. p. 87-101.

KOMOROWSKI, J. et al. Rough sets: a tutorial. In: PAL, S. K.; SKOWRON, A. (Ed.). **Rough fuzzy hybridization**: a new trend in decision making. Singapore: Springer, 1999.

LEI, T. C.; WAN, S.; CHOU, T. Y. The comparison of PCA and discrete rough set for feature extraction of remote sensing image classification: a case study on rice classification, Taiwan. **Computational Geosciences**, Amsterdam, v. 12, n. 1, p. 1-14, 2008.

LIU, H. et al. The oil–gas prediction of seismic reservoir based on rough set and PSO algorithm. In: IEEE INTERNATIONAL SYMPOSIUM ON SIGNAL PROCESSING AND INFORMATION TECHNOLOGY, 7., 2007, Cairo. **Proceedings**… New York: IEEE, 2007. p. 657-662.

NORRIS, M. W.; FAICHNEY, A. K. (Ed.). **SEG Y rev 1 data exchange format**. 2001. Disponível em: <http://www.seg.org/Portals/0/SEG/News%20and%20Resources/Technical%20Standards/seg_y_rev1.pdf >. Acesso em: 27 nov. 2016.

ØHRN, A. **Rosetta technical reference manual**. 2001. Disponível em: <https://pdfs.semanticscholar.org/8fc3/7cdce0a481e3a44fe7fa364e54bd9873c8df.pdf>. Acesso em: 23 nov. 2016.

ONG, H. Y. **Accelerating data-intensive applications using mapReduce**. 2010. 70 f. Dissertação (Mestrado em Ciência da Computação)- The University of Edinburgh, Edinburgh, 2010.

PAL, S. K.; SKOWRON, A. (Ed.). **Rough fuzzy hybridization**: a new trend in decision making. Singapore; New York: Springer, 1999.

PATRÍCIO, C. M. M. M.; PINTO, J. O. P.; SOUZA, C. C. de. Rough sets: técnica de redução de atributos e geração de regras para classificação de dados. In: CONGRESSO

NACIONAL DE MATEMÁTICA APLICADA E COMPUTACIONAL, 28., 2005, São Paulo. **Anais**... São Carlos: Sociedade Brasileira de Matemática Aplicada e Computacional, 2005.

PAWLAK, Z. Rough set. **International Journal of Information and Computer Sciences**, [S. l.], v. 11, n. 5, p. 341-356, 1982.

_____. **Rough set**: theoretical aspects of reasoning about data. Dordrecht: Kluwer Academic Publishers, 1991.

_____. Rough sets and intelligent data analysis. **Information Sciences**, New York, v. 147, n. 1-4, p. 1-12, 2002.

PEREIRA, F. H.; SASSI, R. J. Rough sets and principal components analysis: a comparative study on customer database attributes selection. **African Journal of Business Management**, [S.l.], v. 6, n. 10, p. 3822-3828, 2012.

PIVOTAL. **Pivotal Greenplum database**. 2017. Disponível em: <https://gpdb.docs.pivotal.io/43113/pdf/GPDB43RefGuide.pdf>. Acesso em : 1 fev. 2017.

POSTGRESQL. **PostgreSQL 8.2.23 documentation**. 2006. Disponível em: <https://www.postgresql.org/docs/8.2/static/>. Acesso em: 1 dez. 2015.

RAILSBACK, L. B. **Flat spots and bright spots in seismic data**. 2011. Disponível em: <http://www.gly.uga.edu/railsback/PGSG/PGSGmain.html>. Acesso em: 01 dez. 2016.

ROBINSON, E. A.; TREITEL, S. **Geophysical Signal Analysis**. Englewood Cliffs: Prentice-Hall, 1980.

RODEN, R.; FORREST, M.; HOLEYWELL, R. The impact of seismic amplitudes on prospect risk analysis. **The Leading Edge**, Tulsa, v. 24, n. 7, p. 706-711, 2005.

SCHINELLI, M. C. **Interpretação sísmica para geólogos de petróleo**. 2011. Disponível em: <https://pt.scribd.com/doc/296851265/Interpretacao-Sismica-Para-Geologos>. Acesso em: 1 jan. 2017.

SELVI, S. et al. **A study on computational intelligence techniques to data mining**. 2014. Disponível em: <http://airccj.org/CSCP/vol4/csit42724.pdf >. Acesso em: 23 nov. 2016.

SHAFER, J.; AGRAWAL, R.; MEHTA, M. SPRINT: a scalable parallel classifier for data mining. In: INTERNATIONAL CONFERENCE ON VERY LARGE DATA BASES, 22., 1996, Bombay. **Proceedings**... New York: ACM, 1996. p. 544-555.

SHERIFF, R. E. **Encyclopedic Dictionary of Exploration Geophysics**. 3. ed. Tulsa: Society of Exploration Geophysicists, 1991.

SILVA, P. M. C. e. **Visualização volumétrica de horizontes em dados sísmicos 3D**. 2004. 108 f. Tese (Doutorado)-Pontifícia Universidade Católica do Rio de Janeiro, Rio de Janeiro, 2004.

SONG, J. et al. A faults identification method using dip guided facet model edge detector. In: STEEPLES, D. **SEG Technical Program Expanded Abstracts**, [S.l.: s.n.], 2012. p. 1-5.

THOMAS, J. E. (Org.). **Fundamentos da Engenharia de Petróleo**. Rio de Janeiro: Interciência, 2004.

VASHIST, R.; GARG, M. L. Rule generation based on reduct and core: a rough set approach. **International Journal of Computer Applications**, [S.l.], v. 29, n. 9, p. 1-5, 2011.

YU, Y.; KELLEY, C.; MARDANOVA, I. A pattern recognition-based horizon auto-tracking algorithm. In: INTERNATIONAL CONGRESS OF THE BRAZILIAN GEOPHYSICAL SOCIETY & EXPOGEF, 13., 2013, Rio de Janeiro. **Proceedings**… [S.l.: s.n.], 2013. p. 1612-1616.

ZARANDI, M. H. F.; KAZEMI, A. Application of rough set theory in data mining for decision support systems (DSSs). **Journal of Industrial Engineering**, New York, v. 1, n. 1, p. 25-34, 2008.

ANNEX A – SEG-Y REVISION 2.0 DATA EXCHANGE FORMAT[1]

SEG-Y FILE STRUCTURE

The SEG-Y format is intended to be independent of the actual medium on which it is recorded. For this standard, the terms file and data set are synonymous. Both terms are a collection of logically related data traces or ensembles of traces and the associated ancillary data.

FILE STRUCTURE

Figure 25 illustrates the structure of a SEG-Y file. Following the optional SEG-Y Tape Label, the next 3600 bytes of the file are the Textual File Header and the Binary File Header written as a concatenation of a 3200-byte record and a 400-byte record. This is optionally followed by Extended Textual File Header(s), which consists of zero or more 3200-byte Extended Textual File Header records. The remainder of the SEG-Y file contains a variable number of Data Trace records that are each preceded by a 240-byte Standard Trace Header and zero or more 240-byte Trace Header Extensions. The Trace Header Extension mechanism is the only structural change introduced in this revision and while not strictly backward compatible with prior SEGY formats, it has been carefully designed to have minimal impact on existing SEG-Y reader software. It should be simple for existing software to be modified to detect the presence of the optional trace headers and either process or ignore any Proprietary Trace Header Extensions.

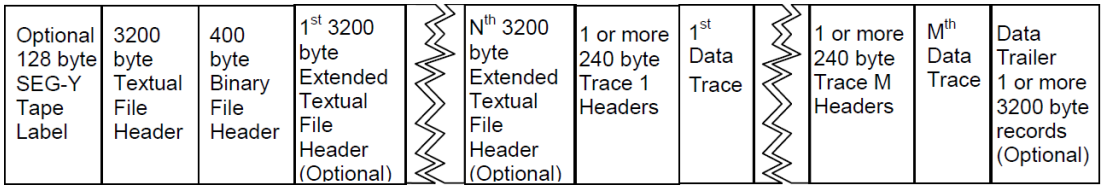


Figure 25. Byte stream structure of a SEG-Y file with N Extended Textual File Header records and M traces records

.

[1] Adapted from: <http:// http://seg.org/Portals/0/SEG/News%20and%20Resources/Technical%20Standards/seg_y_rev2_0-mar2017.pdf>

NUMBER FORMATS

In earlier SEG-Y standards, all binary values were defined as using *big-endian* byte ordering. This means that, within the bytes that make up a number, the most significant byte (containing the sign bit) is written closest to the beginning of the file and the least significant byte is written closest to the end of the file. With SEG-Y rev 2, *little-endian* and *pairwise byteswapped* byte ordering are allowed, primarily for I/O performance. This is independent of the medium to which a particular SEG-Y file is written (i.e. the byte ordering is no different if the file is written to tape on a mainframe or to disk on a PC). These alternate byte orders are identified by examining bytes 3297-3300 in the Binary File Header and apply only to the Binary File Header, Trace Headers, and Trace Samples. All values in the Binary File Header and the SEG defined Trace Headers are to be treated as two's complement integers, whether two, four or eight bytes long, with the exception of the new 8-character Trace Header Extension name, an optional IEEE double precision sample rate, and fields that cannot be negative such as the number of samples per trace. To aid in data recognition and recovery, a value of zero in any SEG or user assigned fields of these headers should indicate an unknown or unspecified value unless explicitly stated otherwise. Trace Data sample values are either integers or floating-point numbers. Signed integers are in two’s complement format. SEG-Y revision 2 adds unsigned integers, 24 and 64 bit integers and IEEE floatingpoint data sample types.

VARYING TRACE LENGTHS

The SEG-Y standard specifies fields for sample interval and number of samples at two separate locations in the file. The Binary File Header contains values that apply to the whole file and the Trace Headers contain values that apply to the associated trace. In SEG-Y, varying trace lengths in a file are explicitly allowed. The values for sample interval and number of samples in the Binary File Header should be for the primary set of seismic data traces in the file. This approach allows the Binary File Header to be read and say, for instance, *this is six seconds data sampled at a two-millisecond interval*. The value for the number of samples in each individual Trace Header may vary from the value in the Binary File Header and reflect the actual number of samples in a trace. The number of bytes in each trace record must be consistent with the number of samples in the Trace Header. This is particularly important for SEG-Y data written to disk files.

Allowing variable length traces complicates random access in a disk file, since the locations of traces after the first are not known without pre-scanning the file. To facilitate the option of random access, a field in the Binary File Header defines a fixed length trace flag. If this flag is set, all traces in the file must have the same length. This will typically be the case for poststack data.

COORDINATES

Knowing the source and receiver locations is a primary requirement for processing seismic data, and knowing the location of the processed data with respect to other data is essential for interpretation. Traditionally seismic coordinates have been supplied as geographic coordinates and/or grid coordinates. SEG-Y accommodates either form. However locations are ambiguous without clear coordinate reference system (CRS) definition. SEG-Y provides the ability to define the CRS used for the coordinates contained within the Binary Header, the Extended Textual Header and the Trace Headers. To avoid confusion, this standard requires that a single CRS must be used for all coordinates within an individual SEG-Y data set. Additionally, the coordinate units must be the same for all coordinates.

TRACE DATA

Trace Data immediately follow their attached Trace Header(s), with the trace data arranged in samples of fixed size (1, 2, 3, 4, or 8 bytes). The format of the data sample is specified in the Binary File Header (bytes 3225–3226). With SEG-Y revision 2, provision has been made via bytes 3297–3300 of the Binary File Header to consistently support littleendian byte ordering or pairwise byte swapping of the Binary File Header and both Trace Headers and Trace Data. The seismic data in a SEG-Y file is organized into ensembles of traces or as a series of stacked traces. When the trace data is organized into ensembles of traces, the ensemble type may be identified (Binary File Header bytes 3229–3230).