Information Sciences Letters

Volume 6
Issue 3 *Sep. 2017*

Article 2

2021

Faults Repairing Analysis Using Rough Sets after Implementation of Labor Force Redistribution Algorithm: A Case Study in Telecom Egypt

O. E. Emam
Information Systems Department, Faculty of Computers and Information, Helwan University, Egypt., amr_it@kfs.edu.eg

M. S. Farhan
Information Systems Department, Faculty of Computers and Information, Helwan University, Egypt., amr_it@kfs.edu.eg

A. A. Abohany
nformation Technology Department, Faculty of Computers and Information, Kafrelsheikh University, Egypt., amr_it@kfs.edu.eg

E. Emam et al.: Faults Repairing Analysis Using Rough Sets after Implementation o

Information Sciences Letters
An International Journal

http://dx.doi.org/10.12785/isl/060301

Faults Repairing Analysis Using Rough Sets after Implementation of Labor Force Redistribution Algorithm: A Case Study in Telecom Egypt

O. E. Emam[1], *M. S. Farhan*[1] *and A. A. Abohany*[2,*]

[1] Information Systems Department, Faculty of Computers and Information, Helwan University, Egypt.
[2] Information Technology Department, Faculty of Computers and Information, Kafrelsheikh University, Egypt.

Received: 8 Jul. 2017, Revised: 25 Aug. 2017, Accepted: 28 Aug. 2017
Published online: 1 Sep. 2017

Abstract: Rough Set Theory (RST) is a data mining technique which is used to deal with vagueness and uncertainty emphasized in decision making. The objective of this paper is to analyze Faults Repairing System (FRS) based on RST before and after applying a suggested algorithm for labor force redistribution. In the first, the indiscernibility relation groups together faults that are indiscernible into equivalence classes, which allowing calculating reducts, FRS analysis in the view of decision regions. Then, Labor Force Redistribution Algorithm (LFRA) is implemented to redistribute faults to malfunctions repairs technicians in the same central or in the cluster according to a set of parameters. Finally, analyzing FRS based on rough sets after applying LFRA. The proposed methodology will be implemented using TE Company as a case study. The results showed that LFRA will improve the accuracy of approximation, maximize the percentage of faults which certainly can be repaired on the same day and minimize the percentage of faults which certainly can't be repaired on the same day.

Keywords: Rough Set Theory, Data Mining, Decision Making, Faults Repairing System, Labor Force Redistribution.

1 Introduction

RST was developed by Pawlak at the beginning of the eighties; it is considered as a new mathematical tool for incomplete data analysis and supports approximations in decision-making [8].

RST is based on the hypothesis that some entities are indiscernible from others if they are classified in the same way regarding their related information; thus, the definitions of indiscernibility relation, lower and upper approximation, and accuracy of the approximation are introduced in RST [23].

RST does not need to give statistical probability distribution of some attributes in advance and do not have to obey any assumptions. RST assumes that entities set analyzed itself imply the knowledge and knowledge is regarded to be a classification ability of the entity. The main objective is to infer rules which can describe each entity classified under which attributes from information system [13].

In recent years, the research and applications on RST have attracted more and more researchers' attention in many areas such as fault diagnosis, image processing, massive data processing, intelligent control systems and others [5,6,10,15,16,24].

Zaras et al. [22] proposed the Dominance-based Rough Set Approach (DRSA) to help the Board of Directors of the Community Futures Development Corporations (CFDC). The CFDC were needed a tool for decision support to identify the projects that were proposed by the contractors and partners of its territory. The DRSA proposal was suitable for the data processing with multiple indicators based on many examples to extract rules related to the proposed model.

Pati et al. [7] presented a new attribute reduction technique, based on directed Minimal Spanning Tree (MST) and RST. In the first, they computed a similarity factor between each pair of attributes using indiscernibility relation. Based on the similarity factors, an attribute similarity set was formed from which a directed weighted graph with vertices as attributes and

* Corresponding author e-mail: amr_it@kfs.edu.eg.

edge weights as the inverse of the similarity factor was created. Then, all possible MSTs of the graph were created. From each tree, iteratively, the most important vertex was included in the reduct set and all its out-going edges are deleted. The process terminates when the edge set is empty, thus producing many reducts.

Sheu et al. [13] analyzed students' misconception based on RST and integrated with an interpretive structural model to match students' degree of two classes. The methodology has provided an effective diagnostic assessment tool for teachers.

Qiu et al. [11] applied the RST to the point cluster and river network selection to meet the requirements of RST. In the first, they formalized the spatial information of entities by the convex hull, triangulated irregular network, Voronoi diagram, etc. In the second, they manually assigned decision attributes to the information table according to condition attributes.

Giudice et al. [3] introduced an application of RST to the real estate area "office units located in Directional District of Naples" and was also integrated with a functional extension so-called "Valued Tolerance Relation" to improve its flexibility. A multiple regression analysis (MRA) was implemented on the same real estate sample with the purpose to compare RST and MRA results.

This paper is organized as follows. Some preliminary concepts about RST are briefly recalled in Section 2. In Section 3, FRS at Telecom Egypt (TE) Company is formulated. FRS is analyzed based on RST in Section 4. Section 5 suggests an algorithm for Labor Force Redistribution (LFR). FRS is analyzed based on RST after applying LFRA in Section 6. Section 7 compares FRS before and after applying LFRA. Section 8 concludes this paper and lists future points.

2 Preliminary Basics about RST

The main concepts of RST are introduced in the following section.

2.1 Information Table

Formally, it is an information system (IS), and can be represented as the following [12,7]:

$$IS = (U, C \cup D, V, f), \tag{1}$$

where

$$U = (x_1, x_2, x_3, \ldots, x_m) \tag{2}$$

is the universe, a non-empty finite set of objects,

$$C = \{c_1, c_2, c_3, \ldots, c_n\} \tag{3}$$

is a non-empty finite set of conditional attributes,

$$D = \{d\} \tag{4}$$

is set of decision attribute,

$$V = \{v_1, v_2, v_3, \ldots, v_n\}, \tag{5}$$

v_i is the range of attribute c_i,

$$f \text{ is a set of } f_{c_i} \text{ information functions,} \tag{6}$$

Each attribute $c_i \in C$ defines an information function

$$f_{c_i} : U \to v_i. \tag{7}$$

2.2 Indiscernibility Relation

According to RST, for every set of attributes $B \subset C$, an indiscernible relation $Ind(B)$ is defined in the following way: two objects x and y are indiscernible by the set of attributes B in C if $b(x) = b(y)$ for every $b \subset B$, where $b(x)$ denotes the value of attribute b for element x. The equivalence class of $Ind(B)$ is called the elementary set in B because it represents the smallest discernible groups of objects. For any element x on the universe U, the equivalence class of x in relation $Ind(B)$ is represented as $[x]_{Ind(B)}$ *or* $[x]_B$ [2,9].

The partitions induced by $Ind(B)$ on the universe U are represented as $U/[x]_{Ind(B)}$ or U/B. Thus, an equivalence relation groups together elements that are indiscernible into classes, providing a partition of the universe of objects.

Consider two sets of attributes $B_1 \subset B_2 \subseteq C$, then $[x]_{B_2} \subseteq [x]_{B_1}$ *and* $U/B_2 \leq U/B_1$ is achieved consequently. That is, a larger set of attributes produces a finer partition of the universe and a smaller set of attributes produces a coarser partition [12].

Consider two sets of attributes $B_1, B_2 \subseteq C$, then $U/B_1 \cap U/B_2 = U/(B_1 \cup B_2)$ and $[x]_{B_1} \cap [x]_{B_2} = [x]_{B_1 \cup B_2}$ is achieved consequently [17].

2.3 Core and Reduct of Attributes

The process of reducing an IS such that the set of attributes of the reduced information system is independent and no attribute can be deleted more without the loss of information from the IS known as reducts [14, 17, 19].

A subset B of the set C is named as a reduct if and only if it includes the following properties [14]:

1. $$U/B = U/C. \tag{8}$$
2. $$U/(B - \{a\}) \neq U/C,\ forall\ a \in B. \tag{9}$$
3. $$[x]_{B-\{a\}} \neq [x]_C,\ foralla \in B. \tag{10}$$

4.A consistency factor of a subset B

$$= \text{a consistency factor of the set } C \tag{11}$$

The core is the necessary element for representing knowledge or rules and is the common part of all reducts,

$$Core\,(IS) = \cap reduct\,(IS)\,. \tag{12}$$

2.4 Upper and Lower Approximations

Let a set $X \in U$, B be an equivalence relation and a knowledge base $K = (U, B)$. The following subsets can be associated [9,12,17,21]:

1.The lower approximation (positive region) set of a set X regarding B is the set of all of the objects, which certainly can be classified with X regarding B, that is,

$$B_*(X) = \cup \{Y \in U/B \mid Y \subseteq X\}\,. \tag{13}$$

2.The upper approximation set of a set X regarding B is the set of all of the objects, which possibly can be classified with X regarding B, that is,

$$B^*(X) = \cup \{Y \in U/B \mid Y \cap X \neq \emptyset\}\,. \tag{14}$$

3.The Boundary Region (BR) set of a set X regarding B is the set of all of the objects, which cannot be decisively classified into X regarding B, that is,

$$BR(X) = B^*(X) - B_*(X)\,. \tag{15}$$

4.The Negative Region (NR) set of a set X regarding B is the set of all of the objects, which cannot be certainly classified into X in regarding B, that is,

$$NR(X) = U - B^*(X)\,. \tag{16}$$

5.A set X is said to be rough set regarding B if its boundary region is non-empty. Otherwise, the set X is crisp regarding B.

6.Four Basic Classes of Rough Sets [9]:

(a)X is roughly B-definable if

$$B_*(X) \neq \emptyset \text{ and } B^*(X) \neq U\,. \tag{17}$$

(b)X is internally B-undefinable if

$$B_*(X) = \emptyset \text{ and } B^*(X) \neq U\,. \tag{18}$$

(c) X is externally B-undefinable if

$$B_*(X) \neq \emptyset \text{ and } B^*(X) = U\,. \tag{19}$$

(d)X is totally B-undefinable if

$$B_*(X) = \emptyset \text{ and } B^*(X) = U\,. \tag{20}$$

7.From the above definitions, the following properties hold [12,17,18]:

(a)

$$B_*(X) \subseteq X \subseteq B^*(X)\,. \tag{21}$$

(b)

$$X \subseteq Y \rightarrow B_*(X) \subseteq B_*(Y)\,. \tag{22}$$

(c)

$$X \subseteq Y \rightarrow B^*(X) \subseteq B^*(Y)\,. \tag{23}$$

(d)

$$B_*(X \cap Y) = B_*(X) \cap B_*(Y)\,. \tag{24}$$

(e)

$$B^*(X \cup Y) = B^*(X) \cup B^*(Y)\,. \tag{25}$$

The following coefficient of the rough set can characterize the accuracy of the approximation [8,14]:

$$\alpha_B(X) = \frac{|B_*(X)|}{|B^*(X)|}\,. \tag{26}$$

Where

$|B_*(X)|$ denotes the cardinality of $B_*(X)$, $|B_*(X)| \neq \emptyset$.

It is clear that $0 \leq \alpha_B(X) \leq 1$. A set X is said to be rough set regarding B if $\alpha_B(X) < 1$; a set X is said to be crisp set regarding B if $\alpha_B(X) = 1$.

2.5 Decision Rules

Decision rules will be denoted by $C(X) \rightarrow D(X)$, where C and D are disjoint sets of the condition and decision attributes, respectively.

A decision rule may be characterized by the following specific definitions [7,14,17,20]:

1.Decision rule support =

$$|C(X) \cap D(X)|\,. \tag{27}$$

2.The strength of the decision rule =

$$\frac{|C(X) \cap D(X)|}{|U|} = \frac{\text{Decision rule support}}{|U|}\,. \tag{28}$$

3.The certainty factor of the decision rule =

$$\frac{|C(X) \cap D(X)|}{|C(X)|}\,. \tag{29}$$

4.If "the certainty factor of the decision rule = 1", then it will be called a certain decision rule; if "0 < the certainty factor of the decision rule < 1", then the decision rule will be viewed as an uncertain decision rule.

5.Coverage factor of the decision rule =

$$\frac{|C(X) \cap D(X)|}{|D(X)|}\,. \tag{30}$$

3 Problem Formulation

Faults repairing the problem at TE Company is formulated into the following sections.

3.1 Geography Area of the paper

This paper has been done in the Sharkia Governorate, located in the northern part of Egypt, in the area operated by TE Company. It has 15 cities and 3885 minor villages with approximately 6,485,412 habitats, scattered on a geographic area of 4,180 km^2.

TE Company is Egypt's main telecommunication company. It was established in 1854. In 1998, it took over the former Arab Republic of Egypt National Telecommunication Organization (ARENTO). It has a fixed-line subscriber base of more than10 million subscribers, which makes it the biggest fixed-line provider in Africa and the Middle East.

3.2 Target Population

The subject of this paper is to analyze FRS based on rough sets before and after applying LFRA and determines whether the faults are repaired the same day for subscribers located in Sharkia Governorate.

TE Company has a fixed-line subscriber base more than 200, 000 subscribers in the branch of Sharkia Governorate.

The branch of Sharkia Governorate has 78 central. A total number of main and sub centrals is 10 and 68, respectively.

Dataset was taken at the period of December 2016 directly from the company's databases.

A total number of dataset records are 795 records.

3.3 Dataset Attributes

Dataset has the following attributes:

1. **Branch ID:** The branch of Sharkia Governorate ID with a numeric data type.
2. **Central ID:** One of the centrals which are located in the branch of Sharkia Governorate with a numeric data type.
3. **Tele No:** Telephone number of the subscriber who faces malfunction with a numeric data type.
4. **Line Type:** one refers to "ground," and two refers to "aerial" with a numeric data type.
5. **Repman ID:** One of the malfunctions repairs technicians that are assigned to the mentioned central and cluster with a numeric data type.
6. **Complaint Date:** The date on which the subscriber submits of the complaint with a date data type.
7. **Complaint Time:** The time on which the subscriber submits of the complaint with a time data type.
8. **Close Date:** The date on which the fault was repaired with a date data type.
9. **Close Time:** The time on which the fault was repaired with a time data type.
10. **Last Test Date-Time:** The date and time on which the fault was tested to check if it is repaired with a date-time data type.
11. **Archived Date-Time:** The date and time on which the complaint was archived with a date-time data type.
12. **Waiting Period:** Period taken to repair the fault with a numeric data type.
13. **Technicians Count:** A total number of the malfunctions repair technicians that are assigned to the mentioned central with a numeric data type.
14. **Faults Count:** A total number of faults that are assigned to the mentioned central with a numeric data type.
15. **In Range:** Yes refers to "Complaint Time" before 3:00 PM; No refers to "Complaint Time" after 3:00 PM with a Boolean data type.
16. **Finished:** decision attribute of the current FRS which has two values:
    (a) **Yes:** refers to the fault is repaired on the same day.
    (b) **No:** refers to the fault isn't repaired on the same day.

4 FRS Analysis and Building Decision Rules Based on RST

The Rosetta (Rough Set Toolkit for Analysis of Data) is a toolkit that implements RST based on rule induction as well as some additional features. It's used the following sections for generating the equivalence classes of the FRS, calculating reduct set of the FRS, identifying decision regions when "Finished" attribute valued "Yes" and "No", and build decision rules of the FRS [1].

4.1 Information Table Construction

First of all, the dataset will be preprocessed and normalized as the following:

1. Some attributes will be deleted because their value doesn't have a direct effect on the decision attribute such as:
    (a) **Branch ID:** All dataset records have the same value ("1").
    (b) **Last Test Date-Time and Archived Date-Time:** Their Values are similar to "Close Date-Time" values.
    (c) **Complaint Date, Complaint Time, Close Date and Close Time:** their values included by "Waiting Period" which represented the difference between "Close Date- Time" and "Complaint Date-Time."

(d)**Tele No:** Domain of codes and has a unique value for every subscriber.
(e)**Repman ID:** Domain of malfunctions repairs technicians that are assigned to the mentioned central and cluster.

The normalized dataset represented the IS of the FRS, which includes the following:

1. Condition attributes are {Central Code, Line Type, In Range, Technicians Count, Faults Count, and Waiting Period}.
2. Decision attribute is {Finished} which has two values. If valued “Yes,” then the fault has been repaired on the same day; if valued “No,” then the fault has not been repaired on the same day.

Central_Code	Waiting Period	Line_Type	In_Range	Technicians Count	Faults Count	Finishied
40	10	2	Yes	0	2	No
40	0	2	No	0	2	No
65	0	1	Yes	1	3	Yes
65	0	1	Yes	1	3	Yes
65	0	1	Yes	1	3	Yes
49	0	2	Yes	1	2	Yes
49	0	2	Yes	1	2	Yes

Fig. 1: FRS representation

Figure 1 represents the FRS in the form of condition attributes and decision attribute (Information Table) for further coming processing.

4.2 Indiscernibility Relation, The Equivalence Classes and Reduct Set of the FRS

Rosetta is used to generating the equivalence classes of the universe based on the indiscernibility relation.

{308, 309, 311, 312, 314, 315, 316, 318, 319, 320, 321, 322, 323, 324, 325, 327, 328, 329, 330, 331, 333}
{310, 313, 317, 326, 332}
{362}
{334}
{585}
{356, 357, 358}
{359}
{361}
{360}

Fig. 2: Equivalence Classes of the FRS

In Figure 2, the indiscernibility relation groups together faults that are indiscernible into equivalence classes to calculate reduct and analyze of the FRS.

Rosetta calculates reduct of the FRS based on the equivalence classes, which is {Central Name, Line Type, In Range, Wait Period}.

4.3 FRS Analysis in the View of Positive Region, Boundary Region, and the Outside Region

Statistics

Approximation
Name: No name

Dimensions

	Number of eq. classes:	Total cardinality:
Universe:	194	795
Upper approximation:	141	664
Lower approximation:	116	258
Boundary region:	25	406
Outside region:	53	131

OK

Fig. 3: Decision Regions When “Finished” Attribute Valued “Yes”

In Figure 3, the “Finished” attribute valued “Yes,” and the following three regions based on the equivalence classes are considered:

1. **Lower Approximation (Positive Region):** Set of faults which certainly can be repaired on the same day.
2. **Boundary Region:** Set of faults which possibly can be repaired on the same day.
3. **Outside Region:** Set of faults which certainly can’t be repaired on the same day.
4. **The accuracy of Approximation** = 258/664=0.388, which means 38.8% of faults with “Finished” attribute valued “Yes” are certainly repaired and 61.2% of its are possible repaired on the same day.

Statistics

Approximation
Name: No name

Dimensions

	Number of eq. classes:	Total cardinality:
Universe:	194	795
Upper approximation:	78	537
Lower approximation:	53	131
Boundary region:	25	406
Outside region:	116	258

OK

Fig. 4: Decision Regions When “Finished” Attribute Valued “No”

In Figure 4, the “Finished” attribute valued “No,” and the following three regions based on the equivalence classes are considered:

1. **Lower Approximation (Positive Region):** Set of faults which certainly can't be repaired on the same day.
2. **Boundary Region:** Set of faults which possibly can't be repaired on the same day.
3. **Outside Region:** Set of faults which certainly can be repaired on the same day.
4. **The accuracy of Approximation** = 131/537=0.243, which means 24.3% of faults with "Finished" attribute valued "No" are certainly can't be repaired and 75.7% of its are possible can't be repaired on the same day.

4.4 Building Decision Rules for the FRS

In Figure 5, Rosetta will be used to build decision rules of the FRS based on the equivalence classes into the form Left Hand Side (LHS)$\rightarrow$ Right Hand Side (RHS) and consider the following for each rule:

1. **LHS Support:** No. of faults that satisfy condition attributes in the LHS of the rule.
2. **RHS Support:** No. of faults that satisfy decision attribute in the RHS of the rule.
3. **RHS Accuracy:** No. of faults that satisfy condition attributes in the LHS of the rule by No. of faults that satisfy decision attribute in the RHS of the rule.
4. **LHS Coverage:** No. of faults that satisfy condition attributes in the LHS of the rule by total No. of faults.
5. **RHS Coverage:** No. of faults that satisfy decision attribute in the RHS of the rule by total No. of faults.

The results in Figure 5 can help the decision makers in predicting the percentage of faults which, certainly repaired, possibly repaired, certainly can't be repaired and possibly can't be repaired in the same day. In rule 9, if conditional attributes "Central Code" = "9", "Line Type" = "2", "In Range" = "Yes", and "Wait Period" = "0", then decision attribute "Finished" = "Yes" or "No" and consider the following for this rule:

1. **LHS Support** = 29.
2. **RHS Support:** 8 for "No" and 21 for "yes".
3. **RHS Accuracy:** 0.275862 for "No" and 0.724138 for "yes".
4. **LHS Coverage** = 0.036478.
5. **RHS Coverage:** 0.041026 for "No" and 0.035 for "yes".

4.5 Operational Research Algorithms in Handling FRS

Emam [4] et al. focused on the solution of fully rough three level large scale integer linear programming problem, in which all decision parameters and decision variables in the objective functions and the constraints are rough intervals, and have block angular structure of the constraints. This paper based on block angular structure where FRS is distributed among several multiple centrals, each central has its own constraints and communicate with other centrals through common constraints which represent the linked point between all centrals.

FRS can be modeled as "Rough Large Scale Integer Linear Programming" problem based on Operational Research (OR) techniques, where:

1. It is distributed among several multiple sub problems (centrals).
2. Decision rules of each central represent the constraints of this central.
3. Constraints of each central are independent of others.
4. No. of faults which can be repaired on the same day are integer values.
5. There are a set of faults which certainly can be repaired on the same day (Lower Approximation) and a set of faults which possibly can be repaired on the same day (Upper Approximation).

The main importance of mapping OR model to rough sets is to deal with large scale decision rules as it without modifications and overcome the complexity of converting it into equations, which allows the decision maker to make rapid decisions.

5 An Algorithm for LFR

The suggested algorithm groups together the nearest centrals into clusters. Then, redistribute faults to malfunctions repairs technicians in the same central or in the same cluster according to:

1. A number of Faults in each central.
2. Average Load.
3. A number of the technician's finished faults up till now.

The following algorithm is repeated daily five times to have in each time a sufficient number of faults to be redistributed. The time slots have the following intervals:

1. At 08:00 AM, the algorithm redistributes all faults which complained before 08:00 AM.
2. At 10:00 AM, the algorithm redistributes all remaining not redistributed faults and faults which complained from 8:00 AM to 10:00 AM.
3. At 12:00 PM, the algorithm redistributes all remaining not redistributed faults and faults which complained from 10:00 AM to 12:00 PM.
4. At 02:00 PM, the algorithm redistributes all remaining not redistributed faults and faults which complained from 12:00 PM to 02:00 PM.
5. At 03:00 PM, the algorithm redistributes all remaining not redistributed faults and faults which complained from 02:00 PM to 03:00 PM.
6. The faults which complained after 03:00 PM is shifted to the next day.

Rule	LHS Support	RHS Suppor
Central_Code(40) AND Waiting Period(10) AND Line_Type(2) AND In_Range(Yes) => Finishied(No)	1	1
Central_Code(40) AND Waiting Period(0) AND Line_Type(2) AND In_Range(No) => Finishied(No)	1	1
Central_Code(65) AND Waiting Period(0) AND Line_Type(1) AND In_Range(Yes) => Finishied(Yes)	3	3
Central_Code(49) AND Waiting Period(0) AND Line_Type(2) AND In_Range(Yes) => Finishied(Yes)	2	2
Central_Code(51) AND Waiting Period(0) AND Line_Type(1) AND In_Range(Yes) => Finishied(Yes)	5	5
Central_Code(51) AND Waiting Period(1) AND Line_Type(1) AND In_Range(Yes) => Finishied(No) OR Finishied(Yes)	2	1, 1
Central_Code(51) AND Waiting Period(10) AND Line_Type(1) AND In_Range(Yes) => Finishied(Yes)	1	1
Central_Code(9) AND Waiting Period(0) AND Line_Type(2) AND In_Range(No) => Finishied(No)	7	7
Central_Code(9) AND Waiting Period(0) AND Line_Type(2) AND In_Range(Yes) => Finishied(No) OR Finishied(Yes)	29	8, 21
Central_Code(9) AND Waiting Period(4) AND Line_Type(2) AND In_Range(Yes) => Finishied(Yes)	1	1

Fig. 5: FRS Decision Rules Using Rosetta

The suggested algorithm can be summarized in the following manner:

Step 1. Set time_slots= [08:00 AM, 10:00 AM, 12:00 PM, 02:00 PM, 03:00 PM].
Step 2. Compute Average_load = total_number_of_faults /total_number_of_technicians.
Step 3. If current_time in time_slots then
Step 4. Get number_of_faults.
Step 5. For i=1 to number_of_faults
Step 6. Get central_code.
Set technician_code= get_technician_from_the_same_central (central_code).
Set technician_load=update_technician_load (technician_code).
Step 7. End For
Step 8. Get remaining_faults.
Step 9. If remaining_faults >0 then
Step 10. For i=1 to remaining_faults
Step 11. Get cluster_code.
Set technician_code= get_technician_from_the_same_cluster (cluster_code).
Set technician_load= update_technician_load (technician_code).
Step 12. End For
Step 13. End If
Step 14. Get remaining_faults.
Step 15. If remaining_faults >0 then
Step 16. For i=1 to remaining_faults
Step 17. Get central_code.
Set technician_code= get_technician_finishied_tasks_central (central_code).
Set technician_load= update_technician_load (technician_code).
Step 18. End For
Step19. End If
Step 20. Get remaining_faults.
Step 21. If remaining_faults >0 then
Step 22. For i=1 to remaining_faults
Step 23. Get cluster_code.
Set technician_code= get_technician_finishied_tasks_cluster (cluster_code).
Set technician_load= update_technician_load(technician_code).
Step 24. End For
Step 25. End If
Step 26. End If
Step 27. Function get_technician_from_the_same_central (central_code)
Step 28. For i=1 to number_of_technicians
Step 29. If technician_load<average_load then
Step 30. Return technician_code.
Step 31. End If
Step 32. End For
Step 33. End Function
Step 34. Function get_technician_from_the_same_cluster (cluster_code)
Step 35. For i=1 to number_of_technicians
Step 36. If technician_load<average_load then
Step 37. Return technician_code.
Step 38. End If
Step 39. End For
Step 40. End Function
Step 41. Function get_technician_finishied_tasks_central (central_code)
Step 42. For i=1 to number_of_technicians
Step 43. If (technician_load - technician_finish) <average_load then
Step 44. Return technician_code.
Step 45. End If
Step 46. End For
Step 47. End Function
Step 48. Function get_technician_finishied_tasks_cluster (cluster_code)
Step 49. For i=1 to number_of_technicians
Step 50. If (technician_load - technician_finish) <average_load then

Step 51. Return technician_code.
Step 52. End If
Step 53. End For
Step 54. End Function
Step 55. Function update_technician_load (technician_ code)
Step 56. Technician_load ++.
Step 57. End function
Step58. Function update_technician_finish (technician_ code)
Step59. Technician_finish ++.
Step60. End function

6 FRS Analysis after Applying LFRA

Now, two condition attributes will be added to the normalized dataset, which represented the IS of the FRS:

1. **Technicians Count in the cluster.**
2. **Faults count in the cluster.**

The previous steps in Section 4 will be repeated and get the following results after applying LFRA.

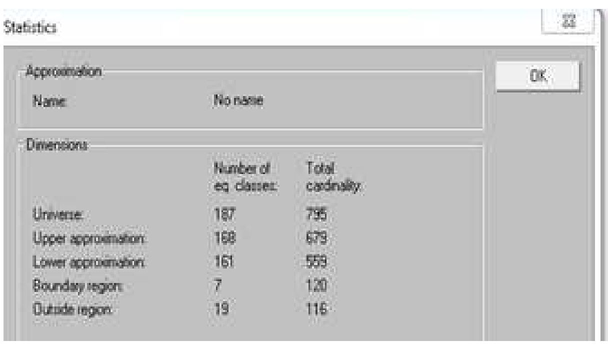

Fig. 6: Decision Regions When “Finished” Attribute Valued “Yes”

In Figure 6, the “Finished” attribute valued “Yes,” and the following three regions based on the equivalence classes are considered:

1. **Lower Approximation (Positive Region):** Set of faults which certainly can be repaired on the same day.
2. **Boundary Region:** Set of faults which possibly can be repaired on the same day.
3. **Outside Region:** Set of faults which certainly can’t be repaired on the same day.
4. **The accuracy of Approximation** = 559/679=0.823, which means 82.3% of faults with “Finished” attribute valued “Yes” are certainly repaired and 17.7% of its are possibly repaired on the same day.

In Figure 7, the “Finished” attribute valued “No,” and the following three regions based on the equivalence classes are considered:

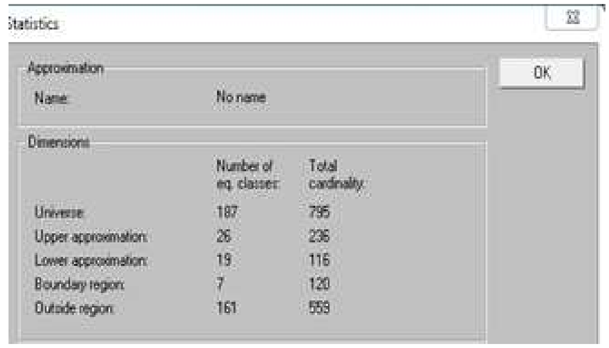

Fig. 7: Decision Regions When “Finished” Attribute Valued “No”

1. **Lower Approximation (Positive Region):** Set of faults which certainly can’t be repaired on the same day.
2. **Boundary Region:** Set of faults which possibly can’t be repaired on the same day.
3. **Outside Region:** Set of faults which certainly can be repaired on the same day.
4. **The accuracy of Approximation** = 116/236=0.491, which means 49.1% of faults with “Finished” attribute valued “No” are certainly can’t be repaired and 50.9% of its are possible can’t be repaired on the same day.

7 Experimental results before and after Applying LFRA

Table 1: Decision Regions When “Finished” Attribute Valued “Yes”

Item	Before LFRA	After LFRA
Lower approximation	258	559
Upper approximation	664	679
Outside region	131	116
Accuracy of approximation	38.8%	82.3%

In Table 1 the “Finished” attribute valued “Yes,” and considered the following:

1. The lower and upper approximations are maximized after applying LFRA.
2. The outside region is minimized after applying LFRA.
3. The accuracy of the approximation is maximized after applying LFRA.
4. The percentage of faults which certainly can be repaired on the same day is maximized.

In Table 2, the “Finished” attribute valued “No,” and considered the following:

Table 2: Decision Regions When "Finished" Attribute Valued "No"

Item	Before LFRA	After LFRA
Lower approximation	131	116
Upper approximation	537	236
Outside region	258	559
Accuracy of approximation	24.3%	49.1%

1. The lower and upper approximations are minimized after applying LFRA.
2. The outside region is maximized after applying LFRA.
3. The accuracy of the approximation is maximized after applying LFRA.
4. The percentage of faults which certainly can't be repaired on the same day is minimized.

8 Conclusion and Points for Future Work

This paper studied FRS based on rough sets before and after applying LFRA. In the first, FRS is analyzed in the view of decision regions based on information tables, indiscernibility relation, reducts and decision rules. Then, LFRA is implemented to redistribute faults to malfunctions repairs technicians in the same central or in the cluster according to a set of parameters. Finally, FRS is analyzed based on rough sets after applying LFRA.

The proposed methodology is implemented using TE Company as a case study to clarify the suggested model. The results showed that LFRA improved the accuracy of approximation, the percentage of faults which certainly can be repaired on the same day is maximized, and the percentage of faults which certainly can't be repaired on the same day is minimized.

However, there are many other aspects, which should be explored and studied such as analyze FRS based on FST and compare the results of RST and FST.

References

[1] Z. Abbas and A. Burney, "A Survey of Software Packages Used for Rough Set Analysis," *Journal of Computer and Communications*, vol.4, pp. 10-18, 2016.

[2] X. Chen, L. Chen, S. Wang, W. Yang and S. Lu, "Research on Destriping Based on Rough Set," *Journal of Computer and Communications*, vol. 3, pp. 8-12, 2015.

[3] V. Giudice, P. Paola, and G. Cantisani, "Rough Set Theory for Real Estate Appraisals: An Application to Directional District of Naples," *Buildings*, vol. 7, no. 12, 2017.

[4] O. E. Emam, E. Fathy, and A. A. Abohany, "An Interactive Model for Fully Rough Three Level Large Scale Integer Linear Programming Problem," *International Journal of Computer Applications*, vol. 155, no. 12, pp. 1-11, 2016.

[5] H. Ho, W. Fann, H. Chiang, P. Nguyen, D. Pham, P. Nguyen and M. Nagai, "Application of Rough Set, GSM, and MSM to Analyze Learning Outcome—An Example Introduction to Education," *Journal of Intelligent Learning Systems and Applications*, vol. 8, pp. 23-38, 2016.

[6] J. Marin, K. Zaras, and B. Boudreau-Trudel, "Use of the Dominance-Based Rough Set Approach as a Decision Aid Tool for the Selection of Development Projects in Northern Quebec," *Modern Economy*, vol. 5, pp. 723-741, 2014.

[7] S. Pati and A. Das, "Constructing Minimal Spanning Tree Based on Rough Set Theory for Gene Selection," *International Journal of Artificial Intelligence & Applications*, vol. 4, no. 1, pp. 159-173, 2013.

[8] Z. Pawlak, "Rough Sets," *International Journal of Computer and Information Sciences*, vol. 11, pp. 341-356, 1982.

[9] A. Pessoa and S. Stephany, "An Innovative Approach for Attribute Reduction in Rough Set Theory," *Intelligent Information Management*, vol. 6, pp. 223-239, 2014.

[10] S. Prema and P. Umamaheswari, "Multitude Classifier Using Rough Set Jelinek-Mercer Naïve Bayes for Disease Diagnosis," *Circuits and Systems*, vol. 7, pp. 701-708, 2016.

[11] J. Qiu, R. Wang, and W. Li, "Use of Rough Sets Theory in Point Cluster and River Network Selection" *Journal of Geographic Information System*, vol. 6, pp. 209-219, 2014.

[12] D. Shanthi and S. Vengataasalam, "Rough Set Approach to Rule Induction Mining," *Asian Journal of Research in Social Sciences and Humanities*, vol. 6, no. 9, pp. 100-114, 2016.

[13] T. Sheu, T. Chen, C. Tsai, J. Tzeng, C. Deng and M. Nagai, "Analysis of Students' Misconception Based on Rough Set Theory", *Journal of Intelligent Learning Systems and Applications*, vol. 5, pp. 67-83, 2013.

[14] T. Slimani, "Application of Rough Set Theory in Data Mining," *International Journal of Computer Science & Network Solutions*, vol. 1, no. 3, pp. 1-10, 2013.

[15] Z. Songbian, "Business Intelligence from Customer Review Management Using Rough Set Model," *International Journal of Advanced Research*, vol. 4, no. 3, pp. 816-824, 2016.

[16] S. Tsumoto, "Mining Diagnostic Rules from Clinical Databases using Rough Sets and Medical Diagnostic Model," *Information Sciences*, vol. 162, pp. 65–80, 2004.

[17] Y.Y. Yao, "Rough Set Approximations: A Concept Analysis Point of View," *Rough Sets: International Joint Conference*, vol. 1, pp. 282-296, 2015.

[18] Y.Y. Yao and Y. She, "Rough Set Models in Multi-granulation Spaces," *Information Sciences*, vol. 327, pp. 40–56, 2016.

[19] Y.Y. Yao, "Rough-Set Concept Analysis: Interpreting RS-Definable Concepts Based on Ideas from Formal Concept Analysis," *Information Sciences*, vol. 346–347, pp. 442–462, 2016.

[20] Y.Y. Yao and B. Zhou, "Two Bayesian Approaches to Rough Sets," *European Journal of Operational Research*, vol. 251, pp. 904–917, 2016.

[21] Y.Y. Yao, "Three-Way Decisions with Probabilistic Rough Sets," *Information Sciences*, vol. 180, no. 3, pp. 341-353, 2010.

[22] K. Zaras, J. Marin and B. Boudreau-Trude, "Dominance-Based Rough Set Approach in Selection of Portfolio of Sustainable Development Projects," *American Journal of Operations Research*, vol. 2, pp. 502-508, 2012.

[23] X. Zhang and W. Xu, “Lower Approximation Reduction in Ordered Information System with Fuzzy Decision,” *Applied Mathematics*, vol. 2, pp. 918-921, 2012.
[24] Q. Zhang, Q. Xie and G. Wang, “A Survey on Rough Set Theory and Its Applications,” *CAAI Transactions on Intelligence Technology*, vol. 11, pp. 1-11, 2016.

Osama Emam is a professor of information science at Faculty of Computers and Information, Helwan University, Egypt. He received the Ph.D. degree in Operations Research at Helwan University. His research interest in the multi-level optimization field.

Marwa Salah Farhan holds a Ph.D. in Information Systems. She is a Lecturer, information systems department, Faculty of Computers and information, Helwan University, Egypt. Her research interests focus on Advanced Database Management, Semantic Web, Big data and data science, and software engineering.

Amr Abohany is a teaching assistant, information technology department, Faculty of Computers and Information, Kafrelsheikh University, Egypt. He received the M.Sc. degree of Information Systems at Helwan University. His research interest in the multi-level large scale optimization and data mining.