AN ENHANCED SOFT SET DATA REDUCTION USING DECISION PARTITION ORDER TECHNIQUE

MOHAMMED ADAM TAHEIR MOHAMMED

Doctor of Philosophy in Computer Science

UNIVERSITI MALAYSIA PAHANG

UNIVERSITI MALAYSIA PAHANG

DECLARATION OF THESIS AND COPYRIGHT

Author's Full Name : ______________________________

Date of Birth : ______________________________

Title : ______________________________

____

____

Academic Session : ______________________________

I declare that this thesis is classified as:

- ☐ CONFIDENTIAL (Contains confidential information under the Official Secret Act 1997)*
- ☐ RESTRICTED (Contains restricted information as specified by the organization where research was done)*
- ☑ OPEN ACCESS I agree that my thesis to be published as online open access (Full Text)

I acknowledge that Universiti Malaysia Pahang reserves the following rights:

1. The Thesis is the Property of Universiti Malaysia Pahang
2. The Library of Universiti Malaysia Pahang has the right to make copies of the thesis for the purpose of research only.
3. The Library has the right to make copies of the thesis for academic exchange.

Certified by:

_______________________ (Student's Signature)

_______________________ (Supervisor's Signature)

____
New IC/Passport Number
Date:

____
Name of Supervisor
Date:

NOTE : * If the thesis is CONFIDENTIAL or RESTRICTED, please attach a thesis declaration letter.

THESIS DECLARATION LETTER (OPTIONAL)

Librarian,
Perpustakaan Universiti Malaysia Pahang,
Universiti Malaysia Pahang,
Lebuhraya Tun Razak,
26300, Gambang, Kuantan.

Dear Sir,

CLASSIFICATION OF THESIS AS RESTRICTED

Please be informed that the following thesis is classified as RESTRICTED for a period of three (3) years from the date of this letter. The reasons for this classification are as listed below.

Author's Name
Thesis Title

Reasons (i)

(ii)

(iii)

Thank you.

Yours faithfully,

____
(Supervisor's Signature)

Date:

Stamp:

Note: This letter should be written by the supervisor, addressed to the Librarian, *Perpustakaan Universiti Malaysia Pahang* with its copy attached to the thesis.

SUPERVISOR'S DECLARATION

We declare that we have checked this thesis, this thesis is adequate in terms of scope and quality for the award of the degree of Doctor of Philosophy in Computer Science.

____

(Supervisor's Signature)

Full Name : DR. RUZAINI BIN ABDULLAH ARSHAH

Position : ASSOCIATE PROFESSOR

Date : AUGUST 2017

STUDENT'S DECLARATION

I hereby declare that the work in this thesis is based on my original work except for quotations and citations which have been duly acknowledged. I also declare that it has not been previously or concurrently submitted for any other degree at Universiti Malaysia Pahang or any other institutions.

____

(Student's Signature)

Full Name : MOHAMMED ADAM TAHEIR MOHAMMED

ID Number : PCC13003

Date : AUGUST 2017

AN ENHANCED SOFT SET DATA REDUCTION USING DECISION PARTITION ORDER TECHNIQUE

MOHAMMED ADAM TAHEIR MOHAMMED

Thesis submitted in fulfillment of the requirements
for the award of the degree of
Doctor of Philosophy in Computer Science

Faculty of Computer Systems & Software Engineering

UNIVERSITI MALAYSIA PAHANG

AUGUST 2017

ACKNOWLEDGEMENTS

In the name of ALLAH, the Most Benevolent, the most Merciful. Praise be to Allah that our grace are successes, per success there is acknowledgement and I dedicate this research to my parents firstly, then my brothers and sisters, and I would also like to thank my friend and my uncle Professor Dr. Siddig Abdul El Aziz.

I would like to express my great pleasure and take this opportunity to write with my shortcake characters which like light in terms of clarity and purity, so the flowing phrases felt that we would not even reach the glory and it is not wishful thinking unless certain of patience are required.

I would like to present my characters and from where i will start to express spring compassion off course to my mother that I shared from her affection despite the distance and despite the likelihood of difficulties. Are full of ears step down modestly, there are people such as diamonds they shine in darkness, thanks to Associate Professor Dr. Ruzaini Bin Abdullah Arshah who helped me to discover the mysterious of the false parameters reduction by her guidance and good opinion without weariness, he has genius solution for problem solving and also he is so humble. Bunched words and phrases are racing in thanking some one not only symbol of humility, and also he is a star burn to enlighten the way for us, and I would like to thanks Associate Professor Dr. Wan Maseri Binti Wan Mohd.

I am also grateful to thanks all who contributed in this research and I ask Allah to sustains them health and wellness.

ABSTRACT

Nowadays, redundant data is one of the open issues due to the rapid development of technologies. This issue is more visible especially in decision-making as the behaviour of such data is more complex and due to the uncertainty during a process of decision making. Besides, the need of extra memory is essential as redundant data makes use of storage and produce redundant copies due to its widespread use. Hence, the soft-set reduction techniques are introduced to assist in reducing storage space by facilitating less number of copies with minimum cost per line or per storage. The benefit of soft-set reduction is to foster the decision making process as well as to enhance the decision's quality. Classification techniques that were previously proposed for eliminating inconsistency could not achieve an efficient soft-set reduction, which affects the obtained solutions; thus producing imprecise result. Furthermore, the decomposition based on previous algorithms could not achieve better parameter reduction in available domain space. The decomposition computational cost made during combination generation can cause machine infinite state as Nondeterministic Polynomial time (NP). The decomposition scenario in Rose's and Kumar's algorithms detects the reduction, but could not obtain the optimal decision. The contributions of this research are mainly focused on minimizing choices costs through adjusting the original classifications by decision partition order. Moreover, this research proposes a decision partition order technique to maintain the original classification consistency. The second contribution is enhancing the probability of search domain of Markov chain model. Furthermore, this research proposes an efficient Soft-Set Reduction accuracy based on Binary Particle Swarm optimized by Biogeography-Based Optimizer (SSR-BPSO-BBO) algorithm that can generate accurate decision for optimal and sub-optimal results. The results show that the decision partition order technique performs up to 50% in parameter reduction, while some algorithms could not obtain any reduction. On the other hand, the proposed Markov chain model could significantly represent the robustness of the proposed reduction technique in making the optimal decision and minimising the search domain by up to 33%. In terms of accuracy, the proposed SSR-BPSO-BBO algorithm outperforms other optimization algorithms by up to 100% in achieving high accuracy percentage of a given soft dataset. In addition, the proposed decision partition order technique has reduced the choices costs and thus improves the original classification consistency. Hence, the proposed technique could efficiently enhance the decision quality. Also, the accuracy of original soft-set optimal and sub-optimal results have been improved using an intelligent SSR-BPSO-BBO algorithm. The computational cost of search domain (space) has been enhanced using proposed Markov Chain Model.

ABSTRAK

Massa kina, pertindihan data adalah satu isu terbuka, hasil dari pembangunan pesat teknologi. Isu ini menjadi lebih jelas, terutama dalam membuat keputusan, disebabkan oleh tingkah laku data ini yang lebih kompleks dan tidak metentu semasa proses membuat keputusan. Lagipun, keperluan memori yang lebih besar menjadi penting untuk ruang storan dan hasil salinan bertindih disebabkan penggunaan yang berleluasa. Jadi, teknik set lembut telah diperkenalkan untuk membantu mengurangkan ruang storan dengan memudah cara melalui pengurangan bilangan salinan dengan meminimumkan kos setiap baris atau setiap storan. Manfaat dari pengurangan set lembut adalah menambahbaik proses membuat keputusan, dan meningkatkan kualiti keputusan. Teknik klasifikasi yang dicadangkan sebelum ini untuk menghapuskan ketidakselarasan tidak berjaya mencapai pengurangan set lembut yang efisien, yang memberi kesan kepada penyelesaian yang dihasilkan; akhirnya mengeluarkan keputusan tidak tepat. Tambahan pula, penguraian berdasarkan algoritma terdahulu tidak mencapai pengurangan parameter yang lebih baik di dalam ruang domain yang tersedia. Kos pengiraan penguraian dibuat semasa penjanaan kombinasi boleh menyebabkan keadaan mesin yang tidak terhingga seperti Nondeterministic Polynomial time (NP). Senario penguraian dalam algoritma Rose dan Kumar dapat mengesan pengurangan, tetapi tidak dapat menghasilkan keputusan yang optimum. Sumbangan kajian ini adalah terutamanya fokus ke atas pengurangan kos pilihan melalui penalaan klasifikasi asal menggunakan susunan pembahagian keputusan. Tambahan pula, kajian ini mencadangkan teknik susunan pembahagian keputusan untuk mengekalkan klasifikasi asal secara konsisten. Sumbangan kedua adalah meningkatkan kebarangkalian domain carian dari Model Rantaian Markov. Tambahan pula, kajian ini mencadangkan ketepatan Pengurangan Set Lembut yang efisien berasaskan Binary Particle Swarm yang dioptimumkan dengan algoritma Biogeography-Based Optimizer (SSR-BPSO-BBO) yang boleh menjana keputusan tepat bagi hasil optimum dan sub-optimum. Hasil dapatan menunjukkan yang teknik susunan pembahagian keputusan mempunyai prestasi yang lebih baik dalam pengurangan parameter sehingga 50% sedangkan ada di kalangan algoritma lain tidak mampu mendapat sebarang pengurangan dalam beberapa senario. Pada masa yang sama, Model Rantaian Markov yang dicadangkan boleh mewakili keteguhan teknik pengurangan dalam membuat keputusan optimum dan mengurangkan domain carian sehingga 33%. Dari aspek ketepatan, algoritma SSR-BPSO-BBO yang dicadangkan mempunyai prestasi lebih baik dari algoritma pengoptimuman lain dalam mencapai peratus ketepatan tinggi sehingga 100% dari mana-mana set data lembut. Seterusnya, teknik susunan pembahagian keputusan yang dicadangkan dapat mengurangkan kos pilihan, dan akibatnya meningkatkan klasifikasi asal secara konsisten. Jadi, teknik yang dicadangkan boleh meningkatkan kualiti keputusan dengan efisien. Akhirnya, ketepatan hasil optimum dan sub-optimum dari Set Lembut asal telah ditingkatkan menggunakan algoritma SSR-BPSO-BBO yang bijak. Kos pengiraan untuk domain (ruang) carian telah ditingkatkan menggunakan Model Rantaian Markov yang dicadangkan.

TABLE OF CONTENT

LIST OF TABLES

LIST OF FIGURES

LIST OF SYMBOLS

BBO	Biogeography Based Optimization
BPSO	Binary Particle Swarm Optimization
DSS	Decision Support System
HPC	Hybrid Parameter reduction Combination
FPC	Framework Parameter reduction Combination
IS	Information System
NNs	Neural Networks
NP	Nondeterministic Polynomial time (complexity)
SSR	Soft Set Reduction

CHAPTER 1

INTRODUCTION

1.1 Overview

Knowledge is a proper representation of summarized information, while information is derived from data. Knowledge management is the process of managing knowledge that contributes in the development of organization and obtains a high quality product through technology to provide benefits to people (Edwards, 2011). Knowledge management is a tool that uses effective collaboration to manage and save the quality of the knowledge in its entirety, then enhances the knowledge through generation, codification, transfer, access and store (Merminod and Rowe, 2012). The success of Knowledge Management (KM) depends on how efficient the knowledge can be organized (Ravindran et al., 2014). In other words, the process of KM is about how data need to be properly processed in order to make meaningful information that can be further summarized to deliver knowledge (King, 2009). In the process of managing knowledge in KM System (KMS), reduction process is used to enhance knowledge generation and simplify the information. Considerable effort has been devoted in developing KMS to capture and manage knowledge through the digital capture, storage and retrievals, either in single location or in multiple distributions. The demand for KMS would greatly reduce the storage and answer the context of cost overrun and their key benefit in making knowledge accessible to users anywhere, anytime, to provide less time complexity and to minimize uncertainty.

Nowadays, redundant data is one of the open issues due to the rapid development in technologies. This issue is more visible especially in decision-making, since the behaviour of such type of data that gives more complexity and uncertainty during a process of decision making. Besides, the need for extra memory is essential as the data makes use of storage and produce redundant copies due to the widespread use.

Thus, it has become a crucial need to reduce such huge amount of data that require substantial original soft data characteristics to improve the storage, which will contribute in improving the searching efficiency of an optimal decision for a given problem scenario. For this reason, the demands of reducing choices, cost, combinations complexity and memory space have encouraged researchers to develop smart techniques to address these issues of optioning optimal solutions, and at the same time these intelligent applications must inherit the characteristic of original soft data (Maji et al., 2002; King, 2009; Rose et al., 2010).

Reductions of soft data set contribute to KMS, since reductions intended to overcome the replications problems which are geographically connected as well as to fit the original sources in a small space instead of using data compressions (Min and Eom, 1994; Gottschalk, 2006; Maier, 2007; Fulmer, 2011; Osei-Bryson et al., 2014). In order to reduce data for the use of KMS, some features of original characteristics need to be specified as a way to assist searching for the best decision. The searching strategy and classification (data mining) need to be determined during the reduction process. Data mining technique is used to extract the descriptions of information, then discover and predict the future expectations from large data sets in the process of knowledge discovery task. Data mining relies on many factors, including the dataset source and the value it contained (Chang et al., 2009). Data mining helps reduction process in discovering knowledge especially in dealing with massive data (Adeli and Hung, 1994).

The issue of data redundancy and the use of its reduction have become a key issue in the area of knowledge management. The success of knowledge management is dependent on the ability to organize and protect knowledge in an efficient manner (Min and Eom, 1994), and hence, these repositories can be managed more efficiently with the help of original characteristics and mining (Yu et al., 2009; Dalkir and Liebowitz, 2011). The question is how to extract these data within their respective relationships. In general, all data are two-dimensional, which is represented in the table of two-dimensional soft data sets. As shown in Figure 1.1, the data sets need to be processed using several steps: finding the original classification, filtering the uncertain data, and finally, getting the desired result.

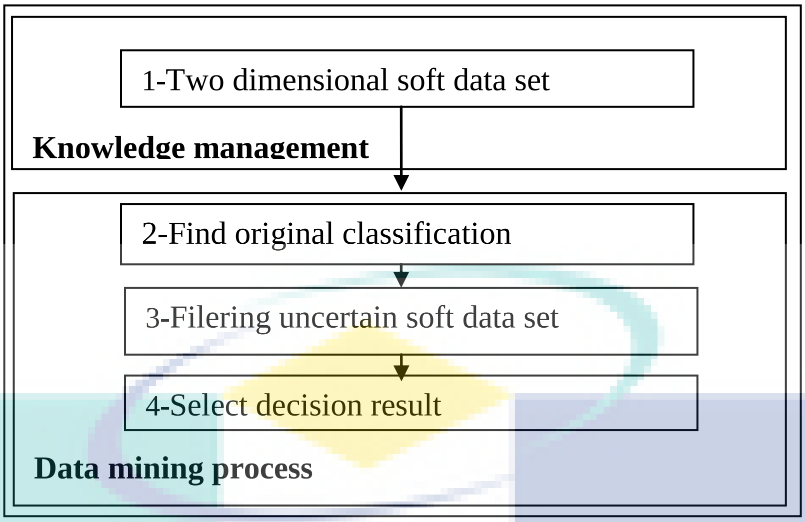

Figure 1.1 Data mining process to achieve reduction in knowledge management

Simultaneously, data mining is performed in discovering knowledge to delve into the massive data, in which the process rely on many factors, such as the data set size and the associated values (Chang et al., 2009). Taking into account the smaller amount of data, the stages involved in the knowledge processing include data extraction, reduction using techniques such clusters and rule associations, and classifications to precisely and accurately predict the result. When the knowledge size is reduced, it is easier to manage and enhance their generation, codification, transfer, access, store and analysis; however, the process of selecting less coefficient data is critical and requires advances in information system (IS) (Merminod and Rowe, 2012).

On the other hand, improving the cost performance of choice is important since economically it assists the customers in the process of decision making, besides significantly saving their money with the obtained optimal choice. This could be achieved when the process taken is rationalizing the cost and simultaneously obtaining an output that has characteristics exactly as the original soft-set. The data volume can be managed by IS as it facilitates data management by rapid processing as long as the key benefits, not only in the process of transmitting and exchanging information anywhere anytime, but their extensive use arise their complexity and uncertain data. The complexity and uncertain data occurs when the information contain repetitions, so it is difficult for a human being to precisely understand their meanings, which could take days to solve such complex problems (Văduva, 2012). Therefore, good information systems (IS) would produce more consistent decisions which give accurate results but at a low cost (Laudon and Laudon, 2004).

Normally, IS generates information with the help of Decision Support System (DSS) that distinguishes between the cost of each choice by comparing their advantages and disadvantages. Using DSS as computer-based model, IS is able to solve the complex decisions problems significantly (Min and Eom, 1994). Besides, DSS can infer reasoning to organize and simplify knowledge management that describe the state of objects in discovering repeated data (Ayyub and Klir, 2006) by using co-relations such as rules or constrains for long-range plan decisions (Min and Eom, 1994). Through the DSS, mathematical models are applied to a computer system as a way to hasten the calculations response time and, with specific characteristics, simplify the decision making process with prompt response (Min and Eom, 1994). However, without a correct and perfect theory, these systems cannot make decision nor give reason as they are normally used just to verify the influence of the data (Akerkar and Sajja, 2010). This yields limited results, and hence the need for algorithm that can overcome the issue in any circumstances and to consistently achieve the needed goals.

Numerous studies have developed and viewed in favor of data reduction to ensure high quality and integrity of data before it can be processed in supporting decision making. In the last decades, several algorithms were developed in the field of data reduction that are aimed at obtaining valuable information. In the case of large data set, the reduction process is very important as the human brain has limitation in performing decision making. Human brain is only able to make decisions based on specific amount of information or choices, but cannot extend the decision beyond the provided information (Del Junco et al., 2010). Consequently, the decision made by humans is often not accurate when dealing with large data set as it is always subjected to uncertainty in the decision making process (Min and Eom, 1994; Chen et al., 2009; Babitha and Sunil, 2010; Asemi et al., 2011;).

Searching strategy is needed in the reduction process to minimize the searching spaces, which eventually will improve data extraction (Fern'ndez-Díaz et al., 2012). Examples of data mining techniques are rule associations, classifications, prediction and regressions. Classification is used to precisely and accurately explore the unique data feature from an original set. To initialize the classification, clusters are need. There are two types of classification method, namely, supervised and unsupervised learning. When the knowledge size is lessened, it is easier to manage and enhance their generation, codification, transfer, access, store, analysis and searching. The process of selecting less coefficient data set is critical in information system (IS) and requires advances data mining techniques (Merminod and Rowe, 2012).

The decision making in KM is not always successful because certain rules are sometimes not carried out (Laudon and Laudon, 2004). Figure 1.2 shows the summarised Decision Making flowchart (Saaty, 1994), which is used to identify correct and incorrect decision in DSS, but DSS itself can neither make decision nor give reasoning (Akerkar and Sajja, 2009). The flowchart represents the decision making process in the information system starting with IS as input, followed by classification model and finally, the mathematical model that distinguish alternatives. The available choices in information system need response from managers or users to make decision; for this reason the decision making strategy support managers and users in order to make decisions.

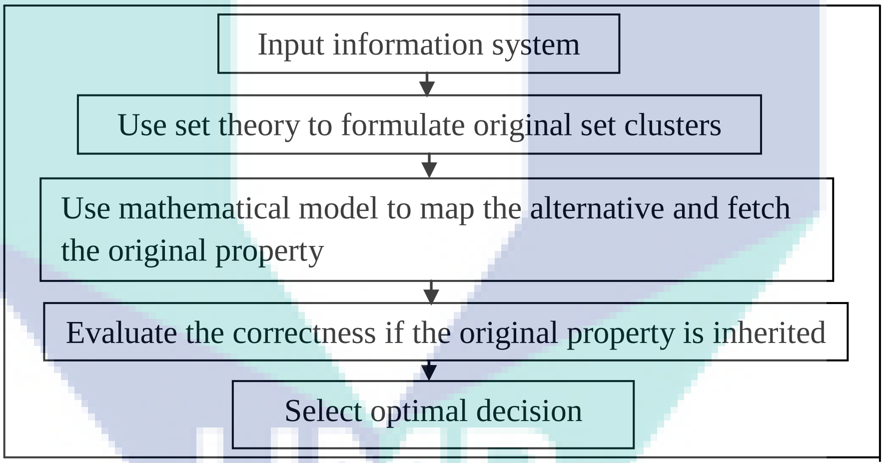

Figure 1.2 Decision making system flow chart

As has been discussed in the preceding paragraphs, in some kind of manual search, such as by trial and error process, could take more time, thus the need for parameter reduction methods. Furthermore, in case of large data in soft-set, it is quite impossible to attain operative reduction; in this regards the problem solving of normal parameter reduction is considered as one of the combinatorial problems. For example, consider that Mr. X wants to buy a house with limited choices that is currently under construction. What kind of house that he/she should buy? It is very important to give Mr X the right and sufficient information about the necessary features and extra features to help him make a correct decision (Puccinelli et al., 2009). The decision cannot be based on emotional influence and bad quality promotions due to their low demand (Puccinelli et al. 2009); consequently, there is a wide range in human daily life where information processing is very important to support them make decision at various levels to ensure accuracy and consistency in decision making (Chen et al., 2009), two types of variables are involved in decision making process, namely, independent and dependent variables; independent variables focus on external factors, while dependent variable focus on main factors in searching for decision quality.

There are several risks to be considered in managing uncertainty in decision making when using several mathematical models. Firstly, rich information is considered as successful complete decision, but incomplete decision can cause a series of risks (Chen et al., 2009). The risk of making decision under uncertainty either exceed boundary domain or contain inconsistency (Jiang et al., 2009). Data consistency is a vital resource to all organization to avoid inconsistency that occurs continuously that everybody faces (Feng and Cagman, 2012). Handling data inconsistency is very important because in reality, there are a lot of real life problems in which uncertain data is involved, such as in the fields of engineering, medical, social, and medical sciences (Maji et al., 2002).

Secondly, obtaining a good decision out of alternatives in business is challenging process. The performance a business will be based on the best quality and lowest prices, as well as in its domestic and the global life in fulfilling customer's satisfaction. In order to fulfil various customers' needs, the features of the available product should be customized and reduced based on their needs. By eliminating the unnecessary features for a particular group of customer could help reduce the cost, hence increase customer satisfaction (Hoque et al., 2013).

Furthermore, in information processing, data integrity and accuracy is very crucial before processing can be done. One of the main issue of data integrity and accuracy is data redundancy or elimination of data similarity. Considering the concept of 'Garbage in garbage out', data similarities in the satisfied boundary should be managed at the initial stage. There are a lot of techniques in managing data similarities such as clusters and AND/OR operations, which are used to minimize the differences in soft data. For example, to manage data similarity in cluster classified data, hierarchical method or partitioning methods can be used, however, the method is more suitable for reducing time complexity, while the partitioning method is more suitable for use if the data is large (Herawan, 2014).

To some extent, not all from the whole amount of data is needed in decision making (Ibrahim and Yusuf, 2012). If excessive information governs the process of decision making, then issues of lengthy processing time and inefficiencies will occur. Thus, in information processing, it is very crucial to process just sufficient data and remove all unnecessary data to reduce processing time.

1.2 Research Background

Numerous fields of research deal with the complexity of uncertain soft data that require efficient techniques to manage and reduce the unnecessary data. The nature of the uncertainty data could be visible in many domains and it varies from one to another. In this regards, the uncertainty problems are frequently handled using some probability theories, such as fuzzy sets, rough sets, soft-sets and other mathematical tools.

In data reduction, there are two main theories are used to manage soft data uncertainty, which are rough set theory and soft-set theory. Rough set theory, as proposed by Pawlak (1983) have boundary that is determined by an approximation in managing uncertain data (Pawlak, 1991). However, this research focuses on soft-set theory as a new technique for managing uncertain data in decision making process. Soft-set theory is also known as binary, basic or elementary system (Molodtsov, 1999). In addition, soft-set may be redefined as the classification of objects in two distinct classes, thus confirming that soft-set can deal with a Boolean-valued information system due to differences of multi-value language in parameters preferences across the world (Min and Dom, 1994). Molodtsov (1999) has pointed out that one of the main advantages of soft-set theory it is free from the inadequacy of the parameterization tools; unlike in the theories of fuzzy set, probability and interval mathematics, in which the inadequacy are visible. Hence, there will be no certain value could be precisely defined to indicate the optimal decision since it is a fuzzy set.

On the other hand, during data reduction process, it is very crucial to ensure that the obtained reduced sets still consist the original properties and attributes of the information (Zhao et al., 2007; Rose et al., 2012). The main objective of reduction is to lessen the number of parameters, and at the same time, attain the property of information in helping the process of decision making (Rose et al., 2012).

In the area of data reduction research, there are various techniques introduced by researchers such as by Maji et al. (2002), Chen et al. (2005), Kong et al. (2008), Mamat et al. (2011), Rose et al. (2010), Rose et al. (2012), and Kumar and Rengasamy (2013). Every soft-set study has verified the influence of parameter exchanges during original combinations generation to search for exact decisions (solutions); the direction of reduction is measured by implicitly or non-implicitly conditions. The noted variation of the implicitly or non-implicitly reductions is that a multiples column yield limited reduction results in the case of uncertainty in the form of non-implicitly. However, there are still issues and challenges in this research area, which provides an opportunity for further research to enhance the existing techniques. For example, reduction technique introduced by Maji et al. (2002) has an issue of sub-optimal problem which causes inconsistency in the obtained results. This problem of inconsistency from Maji was solved by Chen et al. (2005), but the problem of sub-optimal decisions still remain, which induce incorrect and inconsistent decision. Thus, to improve the accuracy of the decision making, Kong et al. (2008) has introduced implicitly reduction technique. However, using Kong's technique, no reduction will be produced if there is no implicitly, and it has complexity even if there is implicitly; thus the reduction proposed by Kong et al. is considered to be partially achieved. The complexity issue of the proposed technique by Kong has been improved by Rose et al. (2010) and Mamat et al. (2011). In addition, Rose et al. (2012) has introduced a technique for identifying the soft-set reduction base on implicitly. In another research, Kumar and Rengasamy (2013) has introduced another technique to reduce the soft data, but without considering implicitly. Thus, it is obvious that as a way to enhance decision making process, the uncertainty parts within the data should be omitted. Hence, data reduction process should take place before performing any decision making process, which would assist in obtaining optimal result. Accordingly, this research is focused on soft data reduction techniques to improve the limitations mentioned by previous researchers, such as Maji et al. (2002), Chen et al. (2005), Kong et al. (2008), Ma et al. (2011), Rose et al. (2010), Rose et al. (2012) and Kumar and Rengasamy (2013).

1.3 Problem Statement

The major problem facing knowledge management is how to control inconsistency of soft-set data, which affects the knowledge derivation, and how to improve the knowledge quality. Previously, the original soft-set theory used to derive choices causes high cost of memory due to processing of searching, which is time consuming. However, decisions made by the existing methods contain a lot of inconsistencies that could affect the actual outcome and increase the risk of the decision taken. Furthermore, the existing parameter reduction algorithms lack the classification ability of the original soft-set (optimal and sub-optimal choices) that could be managed in order to reduce parameters. Besides, issue of false parameters involved in decision making is highly costly and it reduces the accuracy of parameter reduction algorithm.

To sum up the existing gaps within all aforementioned algorithms has motivated the research to develop a new algorithm that could efficiently predicts the optimal and sub-optimal decisions through effective and improved reduction technique. In this research, there are two areas to be explored that are derived from the above issues in order to address the research gaps in normal parameter reduction. The first main issue that need to be addressed is to reduce the data size while ensuring the information is still in a correct form. Then, to find the original features of the information through data classification of optimal and sub-optimal decisions using an efficient classification algorithm.

1.4 Objective of the Study

The objectives of this research are summarized as follows:

i) To design a hybrid technique that could be used to manage uncertain data and maintain its consistency during the parameter reduction process.

ii) To develop a mathematical model that can form the probability of parameter reduction based on adjusted weight vector and Markov Chain model.

iii) To develop an efficient Soft-Set Reduction based Binary Particle Swarm optimized by Biogeography-Based Optimizer (SSR-BPSO-BBO) algorithm that can be utilised in soft-set data classification and optimal decision making processes.

1.5 Scope of Study

The scope of the study are as follows:

i) The scope of this thesis is the use of soft-set theory to reduce the number of soft parameters and objects using the hybrid technique for finite soft data approximations to remove sub objects and determine original data characteristics cluster while maintaining optimal and sub-optimal choices.
ii) Use decompositions and adjusted weight vector to reduce the domain boundary of solutions and select the optimal decisions.
iii) The work is conducted to provide accuracy of decision making in soft-set by establishing an efficient soft-set reduction algorithm.
iv) The proposed algorithms implemented in this study will be using Matlab and Java programs.

1.6 Significant of Study

The proposed algorithm in this study is significant to the body of knowledge which could efficiently maintain the optimal and sub-optimal choices during soft set reduction process. Moreover, the proposed SSR-BPSO-BBO algorithm will efficiently address the issue of obtaining an optimal decision with high classification rate and less error values.

1.7 Organization of Thesis

The rest of this thesis is organized as follows. Chapter 2 describes existing related research and discusses the fundamental concept of rough set theory, rough set-based reduction, the notion of information and relational data base, and the idea of approximation. Chapter 3 describes the proposed techniques, while Chapter 4 describes the analysis and discussions. Finally, the research conclusion and future work are described in Chapter 5.

CHAPTER 2

LITERATURE REVIEW

This chapter is organized as follows. Section 1 presents the introduction of decision making in data mining and soft set theories, while Section 2 describes sets based on mathematical definitions of rough set and soft sets. Section 3 presents a description of previous several soft set algorithms, and finally, Section 4 concludes the chapter.

2.1 Introduction

Significant contributions have been made by previous researchers in reducing soft-set parameters to improve the reduction efficiency of parameters. The reason for the contributions is that inconsistency is a widespread occurrence in soft matrixes. The quality of the soft-set decisions will be increased if there are fewer parameters and if the set contains the required original data. The reduction process is introduced to reduce the number of parameters based on original characteristics, which consequently enhanced memory size, searching, copies and device overflow problem. In data reduction, the data set should be unique and consistent in order to avoid NP-hard and high choices cost (Pawlak and Skowron, 2007).

This chapter presents a general review of the soft-set theory and the characteristics of original data to be exchanged in the generation, which is followed with the discussion on the effect of the reduction on the soft-set and the characteristics of the original parameters to be maintained. Figure 2.1 shows the decision table of a soft-set containing the parameters $(P_1,\ldots,P_n)$ that can be divided, based on support values, into two disjoint groups called condition, which maps the binary value and decision attributes that maintain the objects support $(u_1, u_2, \ldots, u_n)$, where $N = u_1 + u_2 + \ldots + u_n$.

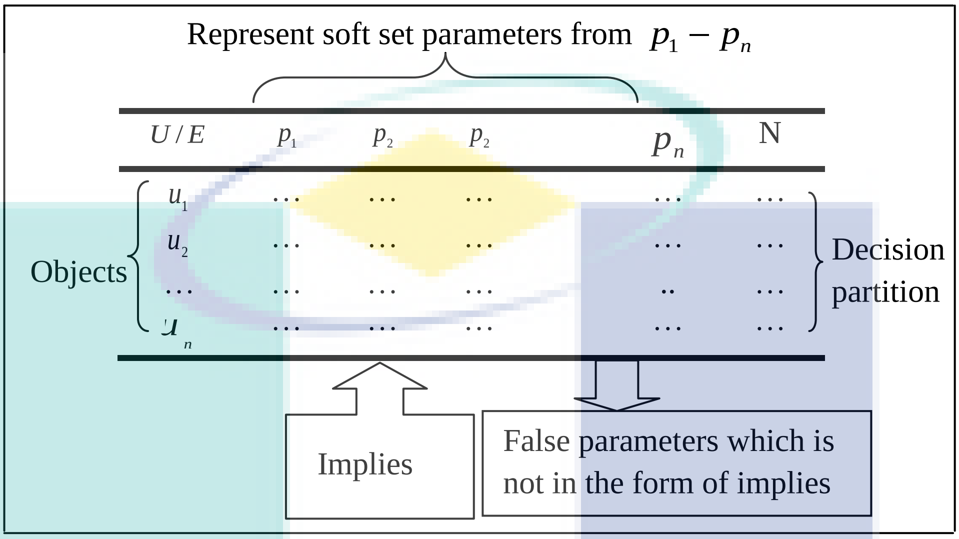

Figure 2.1 Decision table of soft set

Based on Figure 2.1, the challenge in effective decision making is to ensure the unique classifications and appropriate parameter conditions were selected and presented in the decision making. To solve the problem and challenges, many researches have been done in parameter reduction, which highlight the most important parameter correlations in order to optimize the decision making process. However, there is no description in those researches as to when a parameter supports *n* number of records is considered as a reduction; thus, this work attempts to fill the gap and several previous researches were reviewed. For example, the fundamentals of knowledge discovery were carried out in a unique column in order to optimize the memory utilization has been highlighted (Abedjan and Naumann, 2011). Also, traditional data exploration algorithms in parameter reduction results yield insufficient solutions and have certain limitations, such as inaccurate decision discovery (Olson and Delen, 2008). Moreover, reduction of the data boundaries have enhanced time optimization in performing further partition checking based on outstanding results, such as data clustering, data classifications and rule association generation (Ngai et al., 2009). The effect of these techniques is a robust reduction algorithm to support choice classifications in order to ensure the consistency of the choice order (Kantardzic, 2011). However, data classification would require basic configurations and the influence of data classification to generate coefficient of performance for organizing more effective and efficient data categorization (Ji et al., 2011).

Handling uncertain data is very important because, in reality, there are a lot of real life problems in which still involve uncertain data, such as in the fields of engineering, medical, social and medical sciences (Maji, et al., 2002). However, the predicament can be solved using mathematical principles such as rough set ((Kumar et al., 2015; Karaaslan, 2016; Mohamad and Selamat, 2016; Zhan et al., 2017), fuzzy set (Selvachandran and Salleh, 2016), probability (Zhu and Wen, 2010; Deng and Wiebe, 2015; Lee and Wang, 2016), interval (Alkhazale, 2015), and soft-set theory (Cagman and Enginoglu, 2010; Saraf, 2013; Onyeozili and Gwary, 2014; Pant et al., 2015; Tripathy and Arun, 2015; Acıkgoz and Tas, 2016; Awang et al., 2016; Chen and Zou, 2016; Citkin et al., 2016; Gobithaasan, 2016; Solanki, 2016; Ulucay et al., 2016; Yang et al., 2016). Molodtsov (1999) pointed out that one of the main advantages of soft-set theory is that it is free from the inadequacy of the parameterization tools, unlike in the theories of fuzzy set, probability and interval mathematics.

In recent years, research on soft-set theory has been active and great progress has been achieved ((Bakshi et al., 2016; Chen and Zou, 2016; Deli and Cagman, 2016), including that using the fundamental soft-set theory, soft-set theory in abstract algebra and soft-set theory for data analysis, particularly in decision making (Rose et al., 2010; Xu et al., 2014; Dymek and Walendziak, 2015; Feng, 2016; Fertier et al., 2016; Kıvan et al., 2016; Sutoyo et al., 2016; Xu et al., 2016; and Danjuma et al., 2017). The concept of soft-set-based reduction is another area that purportedly supports decision making problems with less involvement of data and attributes by reducing the attributes. An unreduced Boolean-valued database does usually contain large amount of data. Decision making is a very critical task that need to be done by complete but not extraneous information. One practical problem faced is whether all the attributes in the set of attributes are always necessary to preserve a particular property (Zhao et al., 2007), for there might be records that are not significant at all, and vice versa. Thus, one must take into consideration in identifying significant data and use relevant data only for the purpose of decision making. An important issue here is how to reduce the database without removing the crucial data. It is therefore part the intention in this paper is to highlight records and parameters according to its significance and show how it will be dealt with for decision making through the process of spatial reduction. In doing so, it is hoped to scale down the volume of the data such that it is sufficed enough for the process of decision making.

To ensure that sufficient data is available for the process of decision making, reduction of data must be properly addressed. There are cases whereby not only the parameters ought to be considered for the process of reduction, but the object itself is exposed to the process of reduction. Input obtained from the stored data in terms of records and attributes in databases play a major role in the process of decision making. Hence, care must be taken when reducing the data because precision is considered as a major factor in decision making.

Data clustering technique is designed to reduce the number of parameters with similar characteristics (Milligan and Cooper, 1985), where a cluster is the smallest logical amount of storage that can be used to hold data for an organization. In clustering, the relation between elements in the same classes has to be closed and the objective is to predict the features among the objects in order to evaluate each object. To ensure consistency and accuracy of the clusters, the relation between the different classes must be distinct and cannot be closed. However, in reality, the relation in some objects' structure cannot be determined based on the homogeneity or heterogeneity of the objects, but is rather ambiguous or falls in between the two clusters (Hoppner, 1999).

Another common technique used for data reduction is the association rule, which has been used to identify unrelated data with the help of the relations based on support and confidence (Vaidya and Clifton, 2002). In association rule, the values of support factors and confidences were compared with each other to predict the decisions that can be determined by definitions or normalizations. However, weak rules can cause wrong decisions, thus deriving strong rules is crucial; to derive a strong rule domain, the data should be classified with certain criteria and optimized in order to reduce the use of uncertain data (Kantardzic, 2011).

Normally, uncertain data in a soft-set will display the same results regardless of data quality. In this study, it will be shown that a stable original classification is needed to obtain a good reduction result that has quantitative and qualitative coefficient. This implies that it is one of the uncertainties because it contains contradictions, and the occurrence of other forms of false parameters can also result in uncertainty; hence, all uncertainty is noise because it contains inconsistency that reduces the quality of data.

In real situations, the influence of uncertain soft data causes real life problems including a wide range of uncertain data. For example, in field of science, the concept issue has produced methodology to overcome it in engineering, medical, social, medical sciences etc. (Maji et al., 2002) as incorrect result due to noise can generate uncertainty. Thus, the errors resulting from uncertain data need to be handled carefully. These predicaments were dealt with using mathematical principles and soft-set theory to estimate and predict the issues of uncertainty, as used by Molodtsov (1999). A finite uncertain soft data have been solved using soft-set theory, in which every parameter was classified into two 1's and 0's (Molodtsov, 1999), a comprehensive strategy for reducing digital data. The author also suggests that the performance of soft-set reductions is more suitable compared to other reduction strategies.

Duplicate data means that the data will be transferred onto double rows that would result in an inconsistent overall increase in the average of uncertain data, thus increasing choice cost, which consequently serves to decrease the accuracy of the information, and the knowledge of the entire computational domain outlet was directly increased while decreasing the quality of knowledge (Bell, 1982). Tremendous data mining efforts were made since relevant outcomes would be the result of evaluating every single object and eliminating the duplicate data, even though data miming process has performed data cleaning, data integration, data transformation and data reduction. This confirms the argument based on evidence that parameter reduction is dependent on the lowest value and the highest values (Damghani et al., 2009). If the parameters are not beyond the lowest values and the highest value, it needs to integrate the inhomogeneous information by generalization or classification techniques to produce homogeneous information (Ji et al., 2011). Furthermore, to examine the complexity of the outcomes (diversities), consistent and accurate decisions are normally required (Akerkar and Sajja, 2010). Generalization is a mechanism that identifies essential features which run their conditions to mine and match the search (Mitchell et al., 1986); classification that focuses on data categorization is also called generalization since similar objects are organized in classes (Poo et al., 2008). As a result, classification accuracy is the metric of reduction that examine the boundary and decision characteristics to avoid repeated data. For example, suppose in a company the items delivery process is based on email, then repeated email causes repeated items to be sent to customers. Also, by deleting an email the customer has somehow agreed to the loss of his right or at least must wait a certain amount of times (Muller and Freytag, 2005). This question is similar to ambiguities and can be avoided with help of soft-set theory algorithms.

Numerous fields of research deal with the complexity of uncertain data that inquire efficient techniques to manage the reduction process of the unnecessary data. The nature of the uncertainty data could be visible in many domains and it varies from one to another. In this regards there are some of probability theories, such as fuzzy sets, rough sets, soft-sets and other mathematical tools that are frequently used in handling uncertainty problems. In the area of data reduction research, various techniques have been introduced by researchers such as Maji et al. (2002), Chen et al. (2005), Kong et al. (2008), Mamat et al. (2011), Rose et al. (2010), Rose et al. (2012) and Kumar and Rengasamy (2013). Thus, the obvious way to enhance decision making process is to eliminate the uncertainty parts within the data. Hence, data reduction process should take place before performing any decision making process, this will assist in obtaining optimal result. Hence, this research is focused on data reduction techniques.

In addition, during data reduction process, it is very crucial to ensure that the reduced sets obtained still consist the original properties and attributes of the information (Zhao et al., 2007; Rose et al., 2012). The main objective of the reduction is to lessen the number of parameters and maintain the property of information in helping the process of decision making (Rose et al., 2012).

Some researchers have reported soft-set parameters reduction partition modelling. Some previous studies have tried to reduce the data and its ability has great effort, but the vague statements widen the scope of imagination, such as stated by Maji et al. (2002), Chen et al. (2005), Kong et al. (2008), Mamat et al. (2011), Rose et al. (2010), Rose et al. (2012), and Kumar and Rengasamy (2013). Similar investigations that were carried out in reducing implies (Kong et al., 2008; Mamat et al., 2011; Rose et al., 2010) displayed the same reduction results but different using computational strategies. Every soft-set study has verified the influence of parameters exchanges during original combinations generating to search for exact decisions (solutions). Thus, the direction of reduction is measured by implicitly or non-implicitly conditions.

2.2 Optimization and Data Mining Technique

One of the most significant advances in the field of data mining is Neural Networks (NN), inspired by biological neurons in the human brain. At the early phase of NN the concept of rudimentary of NN was mathematically modeled by (McCulloch and Pitts, 1943). Ease of operation, low computational cost, and high performance in addition to accuracy in decision making and classification problems have made this computational tool extraordinarily common over the last years. Of the different types of NNs, the Feedforward Neural Network (FNN) (Fine, 1999) is the easiest and most widely used.

Basically, FNN receives information as inputs on one side and provide outputs from the other side. This connection is normally performed utilizing one-directional links between the neurons in different layers. It is good to mention that FNN is categorized into two types based on number of layers, which are FNN: Single-Layer Perceptron (SLP) (Rosenblatt, 1957) and Multi-Layer Perceptron (MLP) (McCulloch and Pitts, 1943; Werbos, 1974). In case of solving linear problems, SLP type is more compatible since it consists of only a single perceptron. On the other hand, for solving non-linear problems an MLP is more suitable since it has more than one perceptron that could be established in different layers. We can, therefore, highlight the applications of MLPs to be categorized as pattern classification (Melin, Sanchez and Castillo, 2012), data prediction (Guo, Wong and Li, 2012), and function approximation (Gardner and Dorling, 1998). Regarding pattern classification, MLP has the ability to classify data into pre-defined separate classes (Fine, 1999), while in prediction, MPL could perform efficient forecasting of future trends according to present and former data (Guo, Wong and Li, 2012). In addition, MLP can perform function approximation which includes the process of modelling relationships between input variables. It has been proven that MLPs with one hidden layer are able to approximate any continuous or discontinuous function (Hornik, Stinchcombe and white, 1989; Csaji, 2001). Generally speaking, MLPs work similar to the human brain since they are equipped with a learning theory, which offers them the capability to learn from experience. This ability of learning from experience is an essential component with all NNs and it is categorized into two types, which are supervised (Werbos, 1992;Hush and Horne, 1993) and unsupervised (Oja, 2002) learning. In MLPs, most applications are using the standard (Hush and Horne, 1993) or developed Back-Propagation (BP) (Zhang, 2009) algorithms by way of their learning methods, which fit the supervised learning group.

The objective of the learning process is to find the best combination of connection weights and biases in the NN to achieve the minimum error for training and test samples. However, although, thru the learning processes the error frequency of MLP remains continually high for a certain period of time. That could be due to the reason that the learning algorithm leads MLPs to local minima instead of the global minimum. Normally, in gradient-based learning approaches such as BP the issue of trapping within local minima is quite common. The literature demonstrates that heuristic optimization methods are encouraging replacements for gradient-based learning algorithms (Yao, 1993; Branke, 1995). The reason behind that is the stochastic nature of these algorithms enables them to escape local minima better than gradient-based techniques and optimize challenging problems. Heuristic methods also have faster ability in convergence rates to the global minimum compared to BP, as examined in (Gudise and Venayagamoorthy, 2003).

Numerous heuristic optimization methods have been employed to train FNNs, such as Particle Swarm Optimization (PSO) (Mendes, et al., 2002), Genetic Algorithm (GA) (Seiffert, 2001), Ant Colony Optimization (ACO) (Blum and Socha, 2005), and Evolutionary Strategies (ES) (Wienholt, 1993). Hence, the most critical issue of which classification method might be suitable for a specific problem is not easy to be addressed. This concept was proven by a well-known No Free Lunch (NFL) theorem, which states that there is no heuristic algorithm best suited to solving all optimization problems (Adeli, and Hung, 1994; Mirjalili and Hashim, 2010; Boussaid, Lepagnot and Siarry, 2013; Mirjalili et al., 2014).

The BBO algorithm as suggested by (Mirjaliliet et al., 2014) is an Evolutionary Algorithm (EA) that introduces specific evolutionary mechanisms to each individual in a population. EAs such as BBO and GAs can provide more flexible training techniques compared to gradient-based algorithms. The reason behind that is the search space of an MLP is changed for different datasets. In spite of the qualities of GAs for training MLPs, the diversity of evolutionary operators of BBO for each individual potentially allows BBO to overtake a GA in terms of efficiency in classification and decision making problems.

The main motivation behind selecting BBO-based trainer for MLPs is the mutation operator of BBO. Unlike to Swarm Intelligence (SI) techniques (PSO and ACO for instance), EAs mostly have mutation operators, which contributes to their exploitation capability. This possibly enables BO to compete SI techniques in training MLPs as well. A n additional factor that could assist BBO outperform a GA is that there are different mutation constants for each individual in a population of dataset, which usually has a single mutation operator for the whole population. To this end, the essentially different adaptive mechanisms of evolutionary operators and mutations for each individual contribute to BBO in providing diverse exploration and exploitation behaviors when solving different problems.

There is currently insignificant literature focusing on the efficiency of BBO in training MLPs in the field of parameter reduction and decision making based on soft-set technique. The only related work was by Kong (Kong, et al., 2015), in which the particle swarm optimization algorithm was employed to reduce the parameters. However, in the proposed algorithm the standard PSO was utilized, whereby standard PSO is lacking in terms of classification type of problems and does not assist in giving a definite answer or one optimal decision (Mirjalili et al., 2014). Thus, the binary version of PSO (BPSO) is more suitable with parameter reduction issue in soft-set, since it is dealing with two faces probability (0 or 1). Moreover, BPSO needs an optimization process to take place for more accurate results. For this reason, some concepts of optimization process of MLP using BBO are discussed in this section (Figure 2.2).

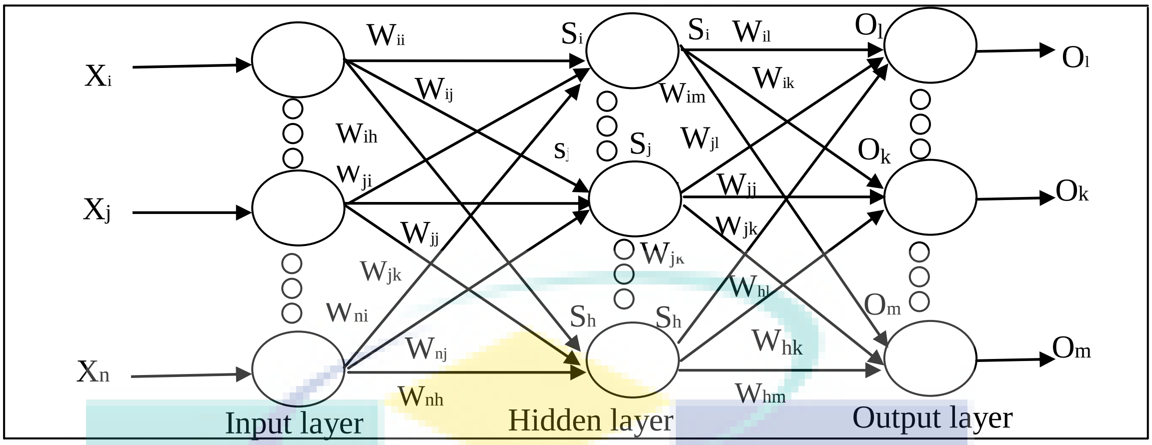

Figure 2.2 An MLP with one hidden layer
Source: McCulloch and Pitts (1943).

Figure 2.2 presents an MLP with three layers, where the number of input nodes is n, the number of hidden nodes is h, and the number of output nodes is m. It is obvious that there is a one-way connection between nodes, which means it is an FNN. The output of the MLP is calculated as follows:

The weighted sums of inputs are first calculated by Eq. (2.1).

$$S_j = \sum_{i=0}^{n} \left(W_{ij} X_i\right) - \theta_j, \qquad j = 1, 2, \dots, h \tag{2.1}$$

where n is the number of the input nodes, W_{ij} represents the weight of the connection from the i_{th} node in the input layer to the j_{th} node in the hidden layer, h_j is the bias (threshold) of the j_{th} hidden node, and X_i shows the i_{th} input. Equation 2.2 is calculating the output of each hidden node.

$$S_j = \text{sigmoid}\left(S_j\right) = \frac{1}{\left(1 + \exp\left(-S_j\right)\right)}, \qquad j = 1, 2, \dots . h \tag{2.2}$$

The final output could be calculated based on the following equation by computing the outputs of hidden nodes.

$$o_k = \sum_{j=0}^{h} \left(W_{jk} S_j\right) - \theta'_k, \qquad k = 1, 2, \dots, m \tag{2.3}$$

where W_{jk} is the connection weight from the J_{th} hidden node to the K_{th}output node, and θ'_k is the bias (threshold) of the kth output node.

$$O_k = \text{sigmoid}(o_k) = \frac{1}{1 + \exp\left(-o_k\right))}, \qquad k = 1, 2, \dots, m \tag{2.4}$$

It is essential to mention that the most significant elements of MLPs are the connection weights and biases. As we can observe from aforementioned equations, the weights and biases are used to identify the final values of output. Thus, the main concept behind training an MLP is to obtain the optimum values for weights and biases as a way to achieve anticipated outputs from certain given inputs of a dataset.

2.2.1 Biogeography-Based Optimization Algorithm

The BBO algorithm was first proposed by Simon (2008). The basic concept BBO algorithm was inspired by biogeography, which relates to the study of biological organisms in terms of geographical distribution. In other words, the basic concept of the geographical distribution over time and space was adopted by Simon as way to introduce BBO algorithm. In BBO algorithm different islands, lands, or even continents over decades, centuries, or millennia could be included in a form of case studies. Thus, various ecosystems (habitats or territories) are explored to find the relationships between different species (habitants) in terms of immigration, emigration, and mutation. Hence, the essential inspiration for the BBO algorithm was come from the evolution of ecosystems in reaching a stable situation while considering different kinds of species (such as predator and prey), and the effects of migration and mutation. In the same way to other EAs which are proposed by Wang (2013; 2014), BBO hires a number of search agents named habitats. These habitats are equivalent to chromosomes in GAs.

The BBO algorithm allocates each habitat a vector of habitants, which is similar to genes in a GA algorithm. This vector is representing the variables of problems. Moreover, the overall fitness of a habitat is indicated by using the defined Habitat Suitability Index (HSI). Thus, higher value of HIS is representing the optimal habitat. Based on the concept of HIS, three main rules performed by habitats in order to progress over time as follows (Ma, 2013):

- Habitants living in habitats with high HSI are more likely to emigrate to habitats with low HSI.
- Habitats with low HSI are more prone to attract new immigrant habitants from those with high HSI.
- Habitats might face random changes in their habitants regardless of their HSI values.

BBO algorithm has utilized these concepts in developing the HSI of all habitats, which increases the number of initial random solutions for a particular problem scenario. Basically, BBO algorithm starts with a random set of habitats and every habitat has n different habitants correlated to the number of variables of an individual problem scenario. In addition, each habitat has its own immigration, emigration, and mutation rates. This simulates the typical geographically separated locations in nature.

Emigration α_k and immigration γ_k are expressed as functions of the number of habitants as follows:

$$\alpha_k = \frac{E \times n}{\mathrm{N}} \tag{2.5}$$

$$\gamma_k = I \times \frac{1 - n}{\mathrm{N}} \tag{2.6}$$

where n is the current number of habitants, N is the allowed maximum number of habitants, which is increased by HSI (the more suitable the habitat, the higher the number of habitants), E is the maximum rate of emigration, and I designates the maximum immigration rate.

2.2.2 Related Optimization Algorithms

It's important to mention that, Simon has proven the BBO algorithm is intelligent to overtake some well-known heuristic algorithms such as PSO, GA, ACO, ES, and Probability-Based Incremental learning (PBIL) (Parmee, 2001; Dasgupta and Michalewicz , 2013) on fourteen benchmark functions and a real problem. He offered BBO as a competitive algorithm in the field of optimization (Simon, 2008).

Regarding GA, it maintains a population of possible solutions to the objective function being optimized. The early group of possible solutions is identified randomly. These possible solutions, called "chromosomes," are permitted to grow over a number of generations. On every generation, the fitness of each chromosome is computed. This fitness will estimate the suitability of possible solution in optimizing the objective function. The preceding generation is shaped via a process of selection and recombination. The chromosomes are elected based on a probabilistic model for recombination based on their fitness. Throughout this process a measurement of how well the chromosomes in achieving the anticipated goal will be identified.

On the other hand, Ant Colony Optimization (ACO) (Blum and Socha, 2005) is used for approximate optimization. The inspiring source of ACO algorithm is from real ant colonies. More precisely, ACO is inspired by the ants' foraging behavior. The indirect communication among ants is considered as an essential component of foraging behavior. This communication is conducted by means of chemical pheromone trails, which allows them to identify the shortest paths between their nest and food sources. This concept of real ant colonies is subjugated in ACO algorithm as a way to solve, for instance, discrete optimization problems.

Furthermore, in Baluja (1994) the population based incremental learning algorithm (PBIL) is proposed and considered as an Estimation of Distribution Algorithms (EDA). PBIL assumes that all the variables of given problem scenario are independent. Also, a probability vector is sustained during each phase of PBIL algorithm. This vector is represented by λ times to achieve λ new solutions. The $\mu \leq \lambda$ best solutions are elected and these have been used to modify the probability vector with a neural networks-inspired rule.

2.3 Set Mathematical Background

The primary question is why we pay high costs in making decisions, and if we had to, it is necessary to use mathematical models to give the most accurate data. The ambiguities appearing in the status of the data affect many other factors, such as data size and data quality, and computational cost increases the solutions boundary leading to overall tedium and confusion; thus, data characteristics identified by proper mathematical formulas usually avoid contradiction (Muller and Freytag, 2005). Ad hoc basis for mathematics is used to make the soft data unique (uniformity). The zones behind the performance of mathematical formula rely on soft-set and rough-set techniques for exchanging data, which minimizes the choices by lower choice rate values, where imperfect information occurs due to indiscernible objects may induce an equivalence class or contain unnecessary features as in soft-set. Parameters are described by binary pairs of information system and the objects mined in equivalent classes in an information system induced dual set of approximations called a lower and an upper approximation (Herawan, 2009c). The authors made a significant contribution by further justifying that reduction based on soft-set theory is equal to rough set theory.

2.4 Rough Set Theory

Earlier rough set theory was introduced by Pawlak in 1982 (Pawlak and Skowron, 2007), which proved that the traditional mathematical concept was crisp and cannot be extended to approximations. However, with modern mathematics, a rough set philosophy performance based on approximations enables the extraction of decisions if approximations occur. The intersection of all the possible reducts is called the core, which represents the most important information of the decision table based on certain Boolean operations.

Rough set extractions filter given samples (objects) by fitting co-efficient preferences from given features (parameters) to find their subsets, in which the filters used relations for knowledge discovery that are dependent on many factors to find unique categories (Polkowski, 2013). Rough set analyses the data (parameters that contain the variables) in the columns of a decision table (Walczak and Massart, 1999; Pawlak et al., 1995). A parameter in the decisions table can be divided or classified into two disjoint groups called boundary (conditions) and acceptable constraints (decision attributes) (Pawlak et al., 1995). Thus, in rough set, each row in a decision table inferred action and result for reasoning knowledge, and its main advantages are as follows (Pawlak, 1998):

i) It does not need additional meta-data.
ii) It provides efficient methods to find accurate data within hidden information.
iii) It allows to summarize original data set without losing knowledge characteristics.
iv) It determines ranges in evaluating the significance of data.
v) It assigns specifications that allow decisions to be automatically generated.

2.4.1 Information System and Indicernibility Relation

Suppose in health care, in the case when the status of a patient is unclear, rough set could take two actions (not crisp) to create and find significant data among roughness to make a decision (Pawlak, 1998). An information system (Pawlak and Skowron, 2007) is a four sub tuple (quadruple) and can be mathematically expressed as $S = (U, A, V, f)$, where U contains objects which include non-empty sets and has a finite set of objects. *A* represents the non-empty attributes that contain a finite elements. $V = \bigcup_{a \in A} V_a$, where V_a represents the mapping conditions that construct the domain (value set) in a given attribute; the mapping is represented by $f : U \times A \rightarrow V$, which is a total function summarized by $f(u,a) \in V_a$, such that for every $(u,a) \in U \times A$. The relations between attributes is called information (knowledge) function. The notion of similar relations (an indicernibility relation) between two objects is utilized in the following definition (Pawlak and Skowron, 2007):

Definition 2.1:

Let $S = (U, A, V, f)$ be an information system and let *B* be any subset of *A* is derived a relation (Pawlak, 1998). Two elements of the universe are similar or equal such that $x, y \in U$ be some collections are said to be configured indiscernible relation in set *B* (Indiscernible by the set of attribute *B* in *S*) if and only if $f(x,a) = f(y,a)$ for every $a \in B$. Obviously, in every finite and non-empty subset elements of *A* are configured with unique indicernibility relation induced equivalence relation which is similar to relations by the set of attribute *B*, therefore the equivalence relation is denoted by $IND(B)$. Again the equivalence relation configured previously induces unique partition. The partition of U formulations induced by U / B in given $S = (U, A, V, f)$ and the equivalence class in the partition U / B containing $x \in U$, denoted by $[x]_B$. Obviously, $IND(B) = \cap_{a \in B} IND\{a\}$.

2.4.2 Set Approximations

The rough set derived the relation between the lower approximation notions and upper approximation notions of a set Pawlak (1982), in which boundary domain is defined as follows (Pawlak and Skowron, 2007):

Definition 2.2:

Let $S=(U,A,V,f)$ represent an information system and let B configure any subset induced by A and let $X \subseteq U$. The set X induced lower and upper approximations that are defined by $\underline{B}(X)=\{x \in U \mid [x]_B \subseteq X\}$ and $\overline{B}(X)=\{x \in U \mid [x]_B \cap X \neq \phi\}$ respectively and summarized as $\underline{B}(X)$ and $\overline{B}(X)$.

The accuracy of approximation induced by set X, $X \subseteq U$, denoted as $\alpha_B(X)$ and measured by $\alpha_B(X)=\dfrac{|\underline{B}(X)|}{|\overline{B}(X)|}$, where $|X|$ denotes the cardinality of X. For empty set ϕ, we define $\alpha_B(\phi)=1$ (Pawlak and Skowron, 2007).

The results generated from approximation measurements show that $0 \leq \alpha_B(X) \leq 1$. If X is a union of some equivalence classes, then $\alpha_B(X)=1$. Thus, the set X is crisp with respect to B, and otherwise, if $\alpha_B(X)<1$, X is rough with respect to B. This measure not only depends on the approximation of X, because by (1) it depends on the approximation of $\neg X$.

A table may have some redundancy, which can be verified using two types of notifications. The first redundancy notification is when the flow of some objects have the same classifications around their objects; one method of reducing the data size of this type of redundancy is to store only the significant objects, where one object represents all the similar objects in each set, as defined in Definition 2.1 when the flow of parameters values in all rows have the same direction. The second redundancy notification is difficult to locate as it requires the classification of the parameters' specific characteristics to be implemented, especially in large data tables. In soft-set sub, columns may be erased while modifying the parameters (re-adjusting the parameters structure) as long as the original classification power coefficient of the information system is not affected (Rose et al., 2012). This concept can be viewed and extended to be included in information systems to distinguish the conditional domains from the decision attributes. After viewing the entire set of attributes, the attributes that are used to describe the objects and do not contain redundancy related to efficiency, such as time-consuming and memory size-consuming, are selected (Zhao et al., 2007).

Attribute re-arrangement is required for reduction process as the objective is to capture the property of the information that describes the original resources.

2.4.3 Reduct and Core

A reduct is a domain that is represented by a minimal set of attributes, which are extracted from specific correction factor for soft rows that preserve the indiscernibility relation. This strategy decreased the number of rows and at the same time increased information quality at the upper density. A core is a small density, which is the reduct of all reductions based on partition support. In order to express high information quality, the density of similar characteristics performance based on core specifications is configured to obtain more precise reduction; hence some preliminary definitions need to be constructed (Pawlak and Skowron, 2007).

Definition 2.3:

Let $S=(U,A,V,f)$, be an information system and let *B* be any subsets of *A* and let a belongs to *B*. We say that a is dispensable (superfluous) in *B* if $U/(B-\{b\})=U/B$, otherwise a is indispensable in *B*. Note that the conditions of a has two statuses, which is dispensable (superfluous) in set *B if* $U/(B-\{b\})=U/B$, otherwise *a* is indispensable in set *B*. The displayed information system should be efficiently displayed, which can be achieved by further dispensing some of the attributes from the system in such a way that the object exchange in the table induced by the original characteristics can still be controlled and discerned, thus maintaining the original characteristic.

Definition 2.4:

Let $S=(U,A,V,f)$ be an information system and let *B* be any subsets of *A*. *B* is called an independent (orthogonal) set if all its attributes are indispensable. Set *B* is independent (orthogonal) if its attributes are relevantly reviewed, and thus the parameters have a strong effect if all of its displayed attributes are indispensable.

Definition 2.5:

Let $S=(U,A,V,f)$ be an information system and let B be any subsets of A. A subset B^* of B is a reduct of B if B^* is independent and $U/B^*=U/B$, generates uniquely independent representation. Thus, a coefficient of a reduct configured is a set of attributes that has been preserved in a partition. It means that to include the original classifications in a reduct which is performed based on a minimal subset of attributes enables the existence of same elements classification in the universe as stated an extension of original classifications to whole set of attributes. Moreover, the attributes that are not a part of a reduct are called superfluous with respect to the original classification induced by the universal elements that changed the original characteristics. While computing equivalence classes is straight forward, the problem is how to find minimal reducts from information systems that are NP-hard. Reducts have several important properties that have been confirmed, one of which is a core, being the representation of unique classifications among the objects that are more than singleton objects.

Definition 2.6:

Let $S=(U,A,V,f)$ represent an information system utilised by attributes and objects. To identify the minimum reduct from set B, let set $B\subset A$. The intersection generated by all reducts ratio is called the core of set B, (the core sensitivity analyses all reducts in the range $Core(B)=\cap \operatorname{Re}d(B)$. Thus, the core of set B is uniformly fixed at the set of all similar classes (indispensable) induced by attribute subsets from set B that drive the same distance in the range. The core is included in every reduct as it is the intersection of all reducts, hence, each element of the core is evaluated to determine whether or not it belongs to some reduct. The core that does not belong to some reduct is independent, and it is the most important property in driving subset of attributes at a constant rate. A correlation is obtained if none of its elements drive the original characteristics while removing subset (classification power of attributes). Decision partition for all objects is represented by core. A core has empirical results which are used to examine and convert data for the purpose of purifying the data in the data analysis process.

2.5 Soft Set Theory

In soft-set theory, the coefficients of Boolean classifications of the reduced data are closer when the space from the induced boundary is small compared to that with larger spacing. Soft theory, as initially defined by Molodtsov (1999), shows that the coefficient of Boolean classifications of the reduced vital digital data has a smaller space compared to the larger spacing boundary induced by interval probabilities. The number of terms in the description of the parameters is limited by the two conditions (“1” and “0”) from which the algorithm can choose (Saraf, 2013). The advantage of soft-set theory using Boolean is that information can be grouped into two classes (“1” and “0”), which are then joined together or separated from each other in the data extraction channel to more efficiently reduce the classes. Various mathematical tools have been used to reduce the data set size in order to generate decisions as functions measuring the relationships, such as probability theory, fuzzy set theory and interval mathematics (Saraf, 2013); the tools were used to fix original specifications then search around the dimensions to find the decision. However, soft-set is another tool that can used to reduce the size of a data set. For instance, Molodtsov (1999) found that the main advantage of soft-set theory is efficiency as the inadequacy of other parametrization tools is eliminated by determining the associated error based on binary relations, unlike in the theories of fuzzy set, probability and interval mathematics. However, soft-set is used to perform binary classification but the challenges are related to finding the relationship among soft data; if the original characteristics are determined, the soft-set will subtract from the original set all lossless choices as the choices are not taken into account. Soft-set methodology filters the original boundary against the candidate keys and is then reduced if it satisfies the original characteristics, which are essentially independent (Demetrovics, 1980). The computation reduces the distance to obtain the similarity in the given pairs; classifications is done using the given rules associated with the related concepts for detecting lower frequency, in which the boundary iteration (sub sets) is used to reduce the bandwidth of the candidate boundary. The decompositions process using rearrangement and optimized parameters reduction gradually improves the candidate boundary. The decomposition process stated in (Ma et al., 2011) search for the parameters that are maximally or minimally associated with objects; the process was expanded in (Kumar and Rengasamy, 2013) to be more flexible for single parameters. However, their time consumption is a real challenge because their complexity can be inferred as (N*M) square, where N is total number of objects while M is total number of rows. In order to reduce matching time for candidate keys, their complexity has to be reduced. By functional dependency, the complexity can be reduced but proper reduction is performed if we check only partitions in filtering false candidates. These dependencies offer better suggestions associated with high quality and the size is lessened, but the challenge is that, in some cases, the dependencies may not contain sufficient information.

Set theory was first introduced by George Cantor in 1983 and has been used to ensure the uniqueness of elements, and to identify elements that are not significant and not correlated. The theory is also used to determine distance between different sets (Pawlak, 1998). A set can be defined as a collection of objects defined as ordered or unordered or logical possibilities and used to specify the object to which collection it belongs (member) (Bernstein, 2005), which solves the problems of memberships in identifying the inverse of an element in every set or it may exist as another set called a sub set (Moschovakis, 2009). Additional information is needed to be assigned to a set to enhance the set performance. The original set or the range of individuals is called the universal set, and if it has one element, it is called a unit set or singleton. The universe equivalence relations induced by partition (Babitha and Sunil, 2010) and proper reduction (subset) contains at least one element which differs from the universe. Set theory is a basis of modern mathematics, which is used in all formal descriptions to manage uncertain data that describe the vagueness (Xu et al., 2010). The set is taken as "undefined", "primitive", or "basic" and the objects are called elements or members. Their concurrences show the detention of set theory to determine the range of sub groups (Fraenkel et al., 1973). This was followed by the example of soft-set reduction introduced by Maji (2002), who stated that the main reduction results are based on AND, OR and NOT. Furthermore, a relative AND and OR arrangement was formulated by Herawan et al. (2009a), which reduces binary soft-set and multi soft-set dimensionality to achieve their reduction coefficients. However, the impact of soft-set algorithm is based on their data exchange.

There are several advantages in the soft-set theory (Saraf, 2013):

i) The researcher can choose any form of parameters that they want because each parameter is associated at most with two values.

ii) Free from parameterization tools compared to fuzzy sets.

iii) When the objects are described they are described with no limited conditions.

iv) It is stable in describing data compared to theory of probability.

v) It simplifies the decision making process, which used two boundary conditions because it is associated with two values, 1s or 0s.

2.5.1 Soft Set Example

Improvements of parameter reduction based on soft reduction that were calculated; for example, the problem of intervals in decision making is how to select possible extension of solutions based on pairs to hold the original characteristics. The extension of interval calculations depends on feasible boundary; if feasible, a boundary is then applied to a prediction that does not contain the right information, but then the extension is not perfect. For this reason soft-set theory can be used where it has a unique feature and free from parameterization tools.

Throughout this section, the use of soft-set theory is illustrated. In soft-set theory, the extension of information system exchangers is well-known. Classified into U that refers to distribution of an initial universe over E, where E elements is a set of parameters, $P(U)$ is the power set of U.

Definition 2.7:

The authors employed the definitions of a pair (F, E) as a soft-set over U, where F is mapping binary values from given parameters such that $F: E \to P(U)$ including the generation of domain. For any original parameter sub set, their relations are described based on its parameterized family as $\varepsilon \in E, F(\varepsilon)$ which in overall is a subset of the universe U. In other words, for $\varepsilon \in E, F(\varepsilon)$ may be considered as the set of ε-elements of the soft-set (F,E) or as the set of ε-approximate elements of the soft-set, instead of a (crisp) set (Refer Maji et al. (2002).

Example 2.1:

As an illustration, consider the computational domain of a soft-set (F, E) determined such as the "attractiveness of automotive promotions" that describes the preference capabilities that Mr. X usually aims for when considering selections to purchase an automobile is one that is cheap and efficient. Assume the parameters in finite volume have fifteen automotive promotion components in the universe U and the propositions are available under construction, $U = \{u_1, u_1, ... u_{15}\}$, and E was collocated as a set of choices (parameters). All variables were inserted in sub set $E = \{p_1, p_2, p_3, p_4, p_5\}$, then the concatenations interpret its meaning such as p_1 stores the values of "large tire" as first parameter, p_2 stores the values of "small tire" as second parameter, p_3 stores the values of "automatic" as third parameter, p_4 stores the values of "manual" as fourth parameter, p_5 stands stores the values of "car status" as the last parameter. The validations of this parameter mapping is derived as $F : E \rightarrow P(U)$ that described the influence of a given "automotive promotions $(.)$", where $(.)$ is used for choices to be filled in to complete the decision. The decision from $p \in E$ is categorized in levels as shown in Figure 2.3.a.

$$
\boxed{\begin{array}{l}
F(p_1) = \{u_1, u_2, u_3, u_4, u_5, u_6, u_7, u_8, u_9, u_{10}, u_{11}, u_{12}, u_{14}, u_{15}\}, \\
F(p_2) = \{u_6, u_7\}, \\
F(p_3) = \{u_5, u_6, u_7, u_8, u_{13}, u_{14}, u_{15}\}, \\
F(p_4) = \{u_3, u_5, u_6, u_7, u_9, u_{10}, u_{11}, u_{12}, u_{13}, u_{14}\}, \\
F(p_5) = \{u_1, u_5, u_6, u_7, u_8, u_{10}, u_{11}, u_{14}, u_{15}\}
\end{array}}
$$

Figure 2.3.a The mapping of parameters of Example 2.1
Source: Rose et al. (2012).

For example, $F(e_2)$ means automotive promotion for manual characteristic. The relationships were inferred among parameters as shown in Figure 2.3.b, which shows the relationship representations that were established to help soft algorithms drop some choices, leaving the significant choices, while some choices may be negligible. Thus, the soft-set (F, E) can be viewed and observed as a collection of approximations that are categorized into similar classes as illustrated in Table 2.1.

$$(F,E)=\left\{\begin{array}{l} p_1=\{u_1,u_2,u_3,u_4,u_5,u_6,u_7,u_8,u_9,u_{10},u_{11},u_{12},u_{14},u_{15}\} \\ p_2=\{u_6,u_7\} \\ p_3=\{u_5,u_6,u_7,u_8,u_{13},u_{14},u_{15}\} \\ p_4=\{u_3,u_5,u_6,u_7,u_9,u_{10},u_{11},u_{12},u_{13},u_{14}\} \\ p_5=\{u_1,u_5,u_6,u_7,u_8,u_{10},u_{11},u_{14},u_{15}\} \end{array}\right\}$$

Figure 2.3.b The soft set of example 2.1

Source: Rose et al. (2012).

Table 2.1 Tabular of soft set parameters values from example 2.1

U/E	p_1	p_2	p_3	p_4	p_5	$f(.)$
u_1	1	0	0	0	1	2
u_2	1	0	0	0	0	1
u_3	1	0	0	1	0	2
u_4	1	0	0	0	0	1
u_5	1	0	1	1	1	4
u_6	1	1	1	1	1	5
u_7	1	1	1	1	1	5
u_8	1	0	1	0	1	3
u_9	1	0	0	1	0	2
u_{10}	1	0	0	1	1	3
u_{11}	1	0	0	1	1	3
u_{12}	1	0	0	1	0	2
u_{13}	0	0	1	1	0	2
u_{14}	1	0	1	1	1	4
u_{15}	1	0	1	0	1	3

The parameters governing the flow of objects are shown in Table 2.1, in which “1” imply the object can be part of a preference, while “0” imply the object is not part of the preference. In the table, the flow of the parameter (preferences) can be dynamically managed by a Boolean-valued information system. The decision partition governing the flow of objects are shown in Table 2.1, in which “1” imply the object can be part of a preference. Similar relations that are found in a soft-set and a Boolean-valued information system is identified as follows:

Proposition 2.1:

If (F, E) is a soft-set over the universe U, then in (F, E) the calculations of a binary-value in information system is determined according to $S = (U, A, V_{\{1,0\}}, f)$ with obtained results similar to soft-set correlation and gives result as shown in the previous table (Rose at el., 2010).

Proof:

Let (F, E) be a soft-set over the universe U, then the mapping configuration is defined as $F = \{f_1, f_2, .., f_n\}$, where the evaluation of the two terms of classification (binary) are assumed as $f : U \longrightarrow V_i$ and $f_i(x) = \begin{cases} 1, & x \in F(e_i) \\ 0, & x \notin F(e_i) \end{cases}$, for $1 \leq i \leq |A|$ between any different configurations. Hence, if $A = E$, for any sub set computed by $V = \bigcup_{a \in A} V_a$, where total parameters exchange $V_{ei} = \{0,1\}$, then a soft-set (F, E) correlations can be considered as a binary-valued information system $S = (U, A, V_{\{1,0\}}, f)$.

From Proposition 2.1, soft-set revises the correlation between classes to make it easier to understand than a binary-valued information system approach, and is suitable in representing and calculating every sub set of preferences from finite soft-set. Thus, a judgement can be made based on a one-to-one correspondence between (F, E) over U and $S = (U, A, V_{\{1,0\}}, f)$.

This section addressed the parameter mapping influences based on soft-set. It shows and focuses on domain boundary of research problems, namely, the soft-set exploration conducted in exact situation as in previous research, which did not completely fill the soft-set reduction within the gap range as discussed in related works. However, the soft-set classification gap adopted in this study is complete.

2.5.2 Soft Set Previous Work

The problem of soft reduction in decision making is on how to select possible extension of solutions that hold the original characteristics. The extension of soft-set is dependent on the feasible boundary; if the boundary does not contain the correct information, the extension will result in low reduction. The extension of this section will discuss the problems of low reduction that will induce low decision quality, or how to minimize the boundary of candidate reduction in several algorithms from previous research. Two common problems were encountered in the algorithms put forward by previous researchers. First, unreduced Boolean data may contain vagueness of the threshold in the database due to the problem of mismatching between the data (Saraf, 2013); in soft-set theory, this problem can be resolved through identifying the relevant data to overcome the vagueness. The second problem is how to validate the outcome, where the original characteristic of the data must be maintained.

Some researchers have examined the soft data relationship based on soft-set theory by focusing on how to increase the effectiveness and efficiency of reduction (Maji et al., 2002; Chen et al., 2005; Kong et al., 2008; Mamat et al., 2011; Rose et al., 2010; Rose et al., 2012; Kumar and Rengasamy, 2013), in which each of the research has its own advantages and inherent limitations in dealing with data uncertainty. One major problem shared by these researchers is that the false frequent parameters are not properly filtered. To solve this problem, the density in parameters should fall within the boundary; density means the number of objects that determines the degree of reduction based on original classification relations (Müller and Freytag, 2005).

The second issue in soft-set relationship is accuracy. Accuracy is a measure of the number of correct values in each object (Müller and Freytag, 2005), to make sure that the quality of knowledge is high. Data accuracy is crucial in ensuring the decision quality (Müller and Freytag, 2005).

2.6 Soft Set Analysis of Previous Work

In order to select optimal solutions, the boundary should be as close as possible and the decision characteristics should be maintained. In the following section will highlight this issue that has been reported in previous works, which include works by Maji et al. (2002), Chen et al. (2005), Kong et al. (2008), Mamat et al. (2011), Rose et al. (2010), Rose et al. (2012) and Kumar and Rengasamy (2013). As an illustration, consider the following case.

Example 2.2:

Let the Cartesian of a soft-set (F, E) represents the information system inferred that gives the reasons by which selected students can choose communication facilities that are available in a university. Assume the boundaries of answers are in Boolean values, and suppose that eighteen students have been surveyed in the universe U with $U = \{u_1, u_2, ... u_{18}\}$, and $E = \{p_1, p_2, p, p_4, p_5, p_6\}$, is a set of parameters that represents the communication facilities used by the student, where p_1 is the inserted value of the parameter for using "email", p_2 is the value of the parameter inserted for using "Facebook", p_3 is the value of the parameter for using "blog", p_4 is the value of the parameter for using in "Friendster's", p_5 is the value of the parameter for using "yahoo messenger" and lastly p_6 is the value of the parameter for using "sms". Consider the computational domain where the mapping is generalized as $F : E \rightarrow P(U)$ and the values of the domain given by "student communication facilities $(.)$, where $(.)$ is to be filled with binary values in parameters $p \in E$.

For example, in Table 2.2, $F(p_2)$ mapped the students that selected Facebook communication and is represented by $\{u_2, u_3, u_4, u_5, u_8, u_9, u_{10}, u_{12}, u_{13}, u_{14}, u_{15}, u_{16}, u_{17}\}$, while $F(p_4)$ mapped the students' use of Friendster and is represented by $\{u_2, u_3, u_4, u_8, u_9, u_{10}, u_{12}, u_{13}, u_{15}, u_{16}, u_{17}, u_{18}\}$. Thus, in overall, the boundary of the parameters can be implemented as approximation, which are represented as shown in Figure 2.4, while the associated Boolean-valued information system is represented in Table 2.2.

$$
(F,E)=\left\{\begin{array}{c}
\text{email} = \{u_1,u_2,u_3,u_4,u_8,u_9,u_{10},u_{11},u_{13},u_{14},u_{15},u_{16},u_{17}\},\\
\text{facebook} = \{u_2,u_3,u_4,u_5,u_8,u_9,u_{10},u_{12},u_{13},u_{14},u_{15},u_{16},u_{17}\},\\
\text{blog} = \{u_1,u_2,u_3,u_4,u_6,u_7,u_8,u_9,u_{10},u_{11},u_{13},u_{15},u_{16},u_{17},u_{18}\},\\
\text{friendsters} = \{u_2,u_3,u_4,u_8,u_9,u_{10},u_{12},u_{13},u_{15},u_{16},u_{17},u_{18}\},\\
\text{ym} = \{u_2,u_3,u_4,u_5,u_6,u_8,u_9,u_{10},u_{11},u_{13},u_{15},u_{16},u_{17},u_{18}\},\\
\text{sms} = \{u_2,u_3,u_4,u_8,u_9,u_{10},u_{13},u_{15},u_{16},u_{17}\}
\end{array}\right\}
$$

Figure 2.4 The soft-set of Example 2.2

Source: Rose et al. (2012).

Table 2.2 Tabular of soft set parameters values from example 2.1

U/E	p_1	p_2	p_3	p_4	p_5	p_6	$f(.)$
u_1	1	0	1	0	0	0	2
u_2	1	1	1	1	1	1	6
u_3	1	1	1	1	1	1	6
u_4	1	1	1	1	1	1	6
u_5	0	1	0	0	1	0	2
u_6	0	0	1	0	1	0	2
u_7	0	0	0	0	1	0	1
u_8	1	1	1	1	1	1	6
u_9	1	1	1	1	1	1	6
u_{10}	1	1	1	1	1	1	6
u_{11}	1	0	1	0	1	0	3
u_{12}	0	1	0	1	0	0	2
u_{13}	1	1	1	1	1	1	6
u_{14}	1	1	0	0	0	0	2
u_{15}	1	1	1	1	1	1	6
u_{16}	1	1	1	1	1	1	6
u_{17}	1	1	1	1	1	1	6
u_{18}	0	0	1	1	1	0	3

2.6.1 Analysis of Soft Set Parameter Reduction in Maji

The idea of operations of soft reduct in decision making was initially proposed by Maji et al. (2002), which is based on OR, AND NOT. In (Maji et al., 2002), the application of soft-set theory solved a decision making problem with the help of Pawlak's rough mathematics (refer Definition 2.5). However, one of the major drawbacks of Maji's proposal is that the identified reduct soft-sets are not similar and yields different maximal weighted value from the identified reduct soft-sets resulting in data inconsistency, which is not wanted in decision making. Therefore, when inconsistencies are present in the process of decision making, one will never be sure which of the different outputs for the same query will be the best choice.

The reduction process forwarded by Maji et al. (2002), detailed in Definition 2.1, uses rough partition based on similarity relation. From the definition of rough partition (refer Definition 2.1), the partition induced by $E = \{p_1, p_2, p_3, p_4, p_5, p_6\}$, is given as U/E= $\{\{u_1, u_5, u_6, u_{12}, u_{14}\}, \{u_2, u_3, u_4, u_8, u_9, u_{10}, u_{13}, u_{15}, u_{16}, u_{u17}\}, \{u_{11}, u_{18}\}, \{u_7\}\}$. From the partition, the optimal decision of objects are $u_2, u_3, u_4, u_8, u_9, u_{10}, u_{13}, u_{15}, u_{16}$ and u_{17}, ,denoted by the maximum value. Meanwhile, the first sub-optimal decision of objects are u_{11} and u_{18} while the second sub-optimal decision of objects are $\{u_1, u_5, u_6, u_{12}, u_{14}\}$. The most inferior object is $\{u_7\}$. From (Maji et al., 2002), the solution has been defined by a specific formula in generating any subset of E such that the sub set will be the reduction if the induced partition is equal to partition U/E based on similar relations. The solution based on the example $D \subset E$, where $D = \{p_1, p_2, p_3, p_4, p_5\}$, is the original dimensionless table as the partition produced is $U/D = \{\{u_1, u_5, u_6, u_{12}, u_{14}\}, \{u_2, u_3, u_4, u_8, u_9, u_{10}, u_{13}, u_{15}, u_{16}, u_{u17}\}, \{u_{11}, u_{18}\}, \{u_7\}\}$. The partition of D with respect to the original partition induced the Boolean-valued table reduction as follows:

Table 2.3 Representation of D which is Maji reduction in Table 2.2

U/E	p_1	p_2	p_4	p_5	$f(.)$
u_1	1	0	0	0	1
u_2	1	1	1	1	4
u_3	1	1	1	1	4
u_4	1	1	1	1	4
u_5	0	1	0	1	2
u_6	0	0	0	1	1
u_7	0	0	0	1	1
u_8	1	1	1	1	4
u_9	1	1	1	1	4
u_{10}	1	1	1	1	4
u_{11}	1	0	0	1	2
u_{12}	0	1	1	0	2

Table 2.3 Continued.

U/E	p_1	p_2	p_4	p_5	$f(.)$
u_{13}	1	1	1	1	4
u_{14}	1	1	0	0	2
u_{15}	1	1	1	1	4
u_{16}	1	1	1	1	4
u_{17}	1	1	1	1	4
u_{18}	0	0	1	1	2

Note that U/D is an invariant to U/E and therefore can be considered as a (rough) attribute reduct of E. The optimal decision of objects are in the set $\{u_2, u_3, u_4, u_8, u_9, u_{10}, u_{13}, u_{15}, u_{16}, u_{17}\}$, denoted by the maximum value of 4. The decision choice for the first sub-optimal decision of object is one of the objects in the set $\{u_5, u_{11}, u_{12}, u_{14}, u_{18}\}$, which is different from the sub-optimal decision derived from E, i.e., $\{u_{11}, u_{18}\}$. Then the second sub-optimal decision of object is now objects in $\{u_1, u_6, u_7\}$ instead of $\{u_7\}$ as it was before the process of reduction. The most glaring issue here is inconsistency as shown in the selection of sub-optimal objects and inferior objects. Basically, in this case, data size has been reduced up to 66% as shown in Table 2.3, but at the expense of consistency.

Summary:

Maji et al. (2002) introduced definitions for generating extractions based on optimal and sub-optimal decision to reduce the size of a large data. The following are the advantages and disadvantages of Maji's techniques.

Advantages:

i) First introduced the AND, OR and NOT operations in soft set theory

ii) Reduce memory size during the extraction of the soft set..

Disadvantages:

i) The result of their extraction in sub optimal choice is not always accurate due to inconsistency of sub optimal solution.

ii) The original characteristic will not be maintained for the above case.

2.6.2 Analysis of Soft Parameter Reduction in Chen's

In another research, Chen et al. (2005) discussed a technique on how to solve the problems encountered in the method suggested by Maji et al. (2002), which can be done by selecting the minimal reduction of the soft-set instead of similarity reduction. Chen pointed out that the variations or result induced by Maji et al. (2002) can be solved with the idea of reduct under rough set theory. However, the terms in the optimal decision partition generally govern the reduction, thus limiting the theory, which suggest that rough set theory cannot be applied directly in reduct under soft-set theory. Chen et al. have ushered in a new idea in the process of decision making based on soft-set theory by introducing the concept of parameter reduction with the hope of overcoming the issue of inconsistencies.

The following section discusses Chen's parameterization reduction technique (Chen et al., 2005). The transformed entries of the table h_{ij} is represented by $f_E(h_i) = \sum_j h_{ij}$ (F, E). A generalized definition of M_E as a collection of objects classified as U in one group, in which the collection showed no change in representing the value of maximum decision counted by f_E. Chen et al. (2005) defined the solution for any $A \subset E$ will be considered as dispensable $M_{E-A} = M_E$ set if induced, otherwise A is called an indispensable set. The maximum collection can be transformed if the original classification is included in their extensions as reduction, such that the set of parameters $A \subset E$ is called a reduction of E if A is indispensable and $M_E = M_E$. However, this derivation by Chen et al. can be considered only in maintaining optimal choice consistency, from which their advantage can be obtained in optimal choices only, which is not considered complete since it excludes maintaining sub-optimal decision. The solution to the problem in maintaining iterations of sub-optimal choices is that their solutions have not been completely addressed. Chen's definition has two types, which are maximum collection and independent.

In the example based on Table 2.1, there is a collection of maximum value $M_E = \{u_5\}$. As in Table 2.1, let a set of parameters be $A = \{P_1, P_2, P_3\}$, thus we have $M_{E-A} = \{p_1, p_2, p_3\} = \{u_2, u_3, u_4, u_8, u_9, u_{10}, u_{13}, u_{15}, u_{16}, u_{17}\}$ and $M_{E-A} = M_E$.

Hence, the set $A=\{p_1, p_2, p_3\}$ can be considered as a parameter reduction of E. Based on Chen's technique, the following binary table (Table 2.4) is derived.

Table 2.4 A binary table of representation of A (a reduction of Table 2.2)

U/E	p_1	p_2	p_3	$f(.)$
u_1	1	0	1	2
u_2	1	1	1	3
u_3	1	1	1	3
u_4	1	1	1	3
u_5	0	1	0	2
u_6	0	0	1	2
u_7	0	0	0	0
u_8	1	1	1	3
u_9	1	1	1	3
u_{10}	1	1	1	3
u_{11}	1	0	1	2
u_{12}	0	1	0	1
u_{13}	1	1	1	3
u_{14}	1	1	0	2
u_{15}	1	1	1	2
u_{16}	1	1	1	3
u_{17}	1	1	1	3
u_{18}	0	0	1	1

As can be seen, the maximum collection is transformed in uniform orthogonal from original Table 2.2 into the table above. Chen et al. (2005) presented the reduction based on maximum collections (exterior), which has successfully maintained the exterior consistency among all objects but the theory failed to maintain interior consistency (sub-optimal decision of objects). The reduction of Table 2.2 could provide reductions as $\{p_1, p_2\}$ or $\{p_2, p_3\}$ because it generates exterior conditions and is independent. For example, the sub-optimal choice from the final form in $\{p_1, p_2\}$ derived the first sub-optimal in the set $\{u_1, u_5, u_{11}, u_{12}\}$, which induced objects decision variant from the original table (Table 2.2) partitioned in the set $\{u_{11}, u_{13}\}$ as it was prior to the process of reduction. However, the last optimal choice in the reduction transformed partition is the set $\{u_6, u_7, u_{18}\}$ instead of inducing original set which was placed in the set $\{u_7\}$. The major drawback was constructed inconsistency as shown in the selection presented in sub-optimal decision of object. Also, the main issue here is inconsistency as shown in the selection of the next sub-optimal decision of objects. Basically, in this case, the data size has been reduced by up to 33%, but at the cost of inconsistency.

Summary:

Chen et al. (2005) attempted to solve Maji et al.'s (2002) reduction issues. The following are the advantages and disadvantages of Chen's technique:

Advantages:

i) Reduce inconsistency from soft set compared to (Maji et al., 2002).

ii) Enhance the soft set reduction compared to (Maji et al., 2002).

Disadvantages:

i) Their sub extractions exclude inconsistency in the sub optimal accuracy which give inaccurate result.

2.6.3 Analysis of Kong et al.'s Soft Parameter Reduction

After the previous soft-set decision making, Kong et al. (2008) used similar spaces between objects as parameter reduction to maintain choice consistency. This arrangement induced new properties that have been applied to choices in solving choice consistency. In this scenario, any implies that occur in a soft-se can be selected to reduce the parameters. Then, Kong et al. defined the new soft-set as a normal parameter reduction, which has been proven to generate accurate result in solving the problems occurred in Chen et al. (2005). The objective of Kong et al.'s technique was to implement new definitions which can be described as one of two types: parameters' degree of importance that was used to analyse and identify the objects supports, and decision partition. Then, the soft decision partition was used to calculate the distances between objects based on maximum parameters. The reduction will be performed if the arrangements of decision partitions at maximum parameters satisfy the definitions of $f_A(P_1) = f_A(P_2) = ... = f_A(P_n)$, implies $C_E = C_{E-A}$. The prosperities generated by normal parameter reduction algorithm improved soft-set reduction such that the distance approximation between objects should be the same while maintaining the optimal and sub-optimal choices. The technique is suitable for removing any implies from soft-set, but their concepts were considered not complete and faced complexity in generating the reductions result. In overall the algorithm is reliable for solving implies conditions.

As shown in the dataset presented in Example 2.2, it is impossible to find a reduct based on Kong et al.'s (2008) normal parameter reduction because the details of the reduction need to maintain the same decision partition. Any parameter deleted from the example will definitely induce levels of approximation, which is different in measuring the decision partition; therefore in governing the reductions, Kong et al.'s definition of normal parameter reduction does not conform to such reduction in the example. Nevertheless, it is hard to find objects conforming the property $f_A(P_1) = f_A(P_2) = \ldots = f_A(P_n)$, while deleting any parameters. Thus, in the case of Kong et al.'s definitions that have been proposed in the normal parameter reduction, as in the examples shown in Table 2.2, there is no normal parameterization reduction at all, and thus the original size of the data was maintained. The main constraint of Kong's (2008) technique in normal parameter reduction is $f_A(P_1) = f_A(P_2) = \ldots = f_A(P_n)$, which sometimes is hard to be derived from the given table. If it is not derivable, then no process of normal parameterization reduction has taken place.

Summary:

The problem encountered in Chen et al. (2005) was successfully solved by Kong et al. (2008), which integrated the idea of maximum decisions in the technique proposed by Chen et al. (2005) to maximum sets induced same decision partition that analyzed the problem of sub-optimal choice and considered optimal choices. It introduces implies definitions to capture false frequent data, which do not always succeed, thus there will be no reduction performed; if it succeeds, there will be partial reduction.

Advantages:

i) Their optimal and sub optimal extraction was correct.
ii) It provides consistency in selecting optimal and sub optimal decisions.
iii) It reduces soft set size.

Disadvantages:

i) If the implies reduction is successful, the data will be partially reduced.
ii) If there is a lot of implies that need to be reduced, no reduction will be performed.

2.6.4 Analysis of Soft Parameter Reduction in Rose (2010)

A framework of decision making based on Maximal Supported Sets, introduced by Rose et al. (2010), was able to overcome the problem of sub-optimal decision faced by Maji et al. (2002) and Chen et al. (2005) by using simpler implies definition. Based on Table 2.2, even the deletion of parameter p_6 maintains the cluster partition $U/A = U/E$, $U/E = \{\{u_1\},\{u_2,u_3,u_4,u_8,u_9,u_{10},u_{13},u_{15},u_{16},u_{17}\},\{u_5\},\{u_6\},\{u_7\},\{u_{11}\},\{u_{12}\},$ $\{u_{14}\},\{u_{18}\}\}$, but supp $A(u) \neq$ supp $A(v)$. Therefore, based on Rose et al. (2010), no reduction occurred with the parameters in Table 2.2.

Summary:

A framework of decision making based on Maximal Supported Sets define implies based on the original cluster. However, the disadvantages of this technique are the same as that faced in the technique suggested by Kong et al. (2008).

Advantages:

i) Their optimal and sub extraction was correct.
ii) It provides consistency in selecting optimal and sub optimal decision.
iii) It reduces soft set size.
iv) It introduces easy implies definitions.
v) The complexity of sub set combinations are not enhanced.

2.6.5 Analysis of Soft Parameter Reduction in Rose (2012)

Hybrid reduction was introduced by Rose et al. (2012), in which row and column reductions are applied. They have developed row correlations that are maximally and minimally supported by parameters and are able to overcome similar false non-implies reductions in addition to implies conditions, which consequently enhanced the framework in decision making by raw reduction (Rose et al., 2010). The hybrid reduction highlights the situation where any object will be deleted if it satisfies maximally or minimally supported by parameter condition, then for any parameter satisfies zero significant will also be deleted.

For the objects found in Table 2.2, the hybrid reduction deleted objects $u_2, u_3, u_4, u_8, u_9, u_{10}, u_{13}, u_{15}, u_{16}, u_{17}$ because they contain maximum ultimate support, and then it induced significant zero column in p_6, hence this will also be deleted. Based on hybrid reduction, the data size in Table 2.2 is reduced by up to 37% as shown in Table 2.5.

Table 2.5 Shows the reduction of table 2.2 based on Hybrid reduction algorithm

U/E	p_1	p_2	p_3	p_4	p_5	$f(.)$
u_1	1	0	1	0	0	2
u_5	0	1	0	0	1	2
u_6	0	0	1	0	1	2
u_7	0	0	0	0	1	1
u_{11}	1	0	1	0	1	3
u_{12}	0	1	0	1	0	2
u_{14}	1	1	0	0	0	2
u_{18}	0	0	1	1	1	3

Summary:

Hybrid reduction reduced the soft set size by enhancing the reduction in (Rose et al., 2010). The following are the advantages and disadvantages of hybrid reduction technique.

Advantages:

i) Their optimal and sub extraction was correct.

ii) It provides consistency in selecting optimal and sub optimal decision.

iii) It reduces soft set size.

i) It has row reduction.

Disadvantages:

i) If there is no object that is maximally or minimally supported by parameters, then the reduction will not be improved as that achieved by Kong et al. (2008), Rose et al. (2010) and Mamat et al. (2011) in terms of implies reduction.

ii) If there is non-implies which are not in the form of (i), there will be no reduction.

ii) The complexity of sub set combinations are not enhanced.

2.6.6 Analysis of Soft Parameter Reduction in Ma (2011)

A new efficient normal parameter reduction algorithm was introduced by Mamat et al. (2011) to improve Kong et al.,'s (2008), complexity which assigned the implies classifications to sub parameters based on parameter support that induced multiples of N, where N is total parameters. Before assigning the implies classifications by default, any parameter maximally or minimally supported by objects will be automatically forwarded to the reduction set. Then, the reduction set from original parameters is reduced to become the candidate domain. Finally, the searching process checks the implies conditions in candidate domain for every set such that all objects supports are the same and then reduced, which consequently improves the complexity of implies introduced by Kong et al. (2008).

Thus, based on data set in Table 2.2, this algorithm results in no reduction, and the disadvantages are the same as faced by Kong et al. (2008). The advantages of this algorithm is that it enhances the complexity of Kong's algorithm and reduces the candidate boundary.

Advantages:

i) Their optimal and sub extraction was correct.

ii) It provides consistency in selecting optimal and sub optimal decision.

It reduces soft set size and removes the complexity induced by (Kong et al. 2008).

Disadvantages:

i) If the implies have not taken place, then the reduction will not be improved .

ii) The complexity of sub set combinations in some cases are not enhanced based on single columns

2.6.7 Analysis of Soft Parameter Reduction in Kumar (2013)

The idea behind the algorithm suggested by Kumar and Rengasamy (2013) is to delete every parameter in a soft-set and, after checking, reduce the set if the remaining parameters generates the support cluster similar to that of the original soft-set. This algorithm has a quick decision in deciding whether or not the parameter can be reduced.

Based on Table 2.2, the parameters support cluster of the original soft-set is:

$$\{\{u_1,u_5,u_6,u_{12},u_{14}\},\{u_2,u_3,u_4,u_8,u_9,u_{10},u_{13},u_{15},u_{16},u_{17}\},\{u_7\},\{u_{11},u_{18}\}\} \quad (2.1)$$

The main problem faced in Kumar and Rengasamy's algorithm is that it does not focus on implies reduction if the parameters are not in one-dimensional representations. As shown in Equation (2.1), in determining the decision partition, the algorithm starts from first record and groups similar objects, then continues to next, and the process continues until last record is reached. However, this process excludes the non-implies that are included in the group of parameters.

Equation (2.1) is different from the original soft-set cluster. After deleting parameter (p_1), the support cluster becomes:

$$\{\{u_1,u_7,u_{14}\},\{u_2,u_3,u_4,u_8,u_9,u_{10},u_{13},u_{15},u_{16},u_{17}\},\{u_5,u_6,u_{11},u_{12}\},\{u_{18}\}\} \quad 2.2)$$

Eq. (2.2) is different from the original soft set cluster. After deleting parameter (p_2), the support cluster is:

$$\{\{u_1,u_6\},\{u_2,u_3,u_4,u_8,u_9,u_{10},u_{13},u_{15},u_{16},u_{17}\},\{u_5,u_7,u_{12},u_{14}\},\{u_{11},u_{18}\}\} \quad (2.3)$$

Eq. (2.3) is different from the original data set cluster. After deleting the parameter (p_3), the support cluster becomes:

$$\{\{u_1,u_6,u_7\},\{u_2,u_3,u_4,u_8,u_9,u_{10},u_{13},u_{15},u_{16},u_{17}\},\{u_5,u_{11},u_{12},u_{14},u_{18}\}\} \quad (2.4)$$

Eq. (2.4) is different from the original data set cluster. After deleting the parameter (p_4), the support cluster becomes:

$$\{\{u_1,u_5,u_6,u_{14},u_{18}\},\{u_2,u_3,u_4,u_8,u_9,u_{10},u_{13},u_{15},u_{16},u_{17}\},\{u_7,u_{12}\},\{u_{11}\}\} \quad (2.5)$$

Eq. (2.5) is different from the original data set cluster. After deleting parameter (p_5), the support cluster can be expressed as:

$$\{\{u_1,u_{11},u_{12},u_{14},u_{18}\},\{u_2,u_3,u_4,u_8,u_9,u_{10},u_{13},u_{15},u_{16},u_{17}\},\{u_5,u_6\},\{u_7\}\} \quad (2.6)$$

Eq. (2.6) is different from the original dataset cluster. The cluster supports after deleting parameter (p_6), is given as:

$$\{\{u_1,u_5,u_6,u_{12},u_{14}\},\{u_2,u_3,u_4,u_8,u_9,u_{10},u_{13},u_{15},u_{16},u_{17}\},\{u_7\},\{u_{11},u_{18}\}\} \quad (2.7)$$

Equation (2.7) is the same as the original data set cluster. As shown in Table 2.6, it shows that the data size in Table 2.2 is reduced by up to 17 % .

Table 2.6 Data size in Table 2.2 reduction in parameterization reduction

U/E	p_1	p_2	p_3	p_4	p_5	$f(.)$
u_1	1	0	1	0	0	2
u_2	1	1	1	1	1	5
u_3	1	1	1	1	1	5
u_4	1	1	1	1	1	5
u_5	0	1	0	0	1	2

Table 2.6 Continued.

U/E	p_1	p_2	p_3	p_4	p_5	$f(.)$
u_6	0	0	1	0	1	2
u_7	0	0	0	0	1	1
u_8	1	1	1	1	1	5
u_9	1	1	1	1	1	5
u_{10}	1	1	1	1	1	5
u_{11}	1	0	1	0	1	3
u_{12}	0	1	0	1	0	2
u_{13}	1	1	1	1	1	5
u_{14}	1	1	0	0	0	2
u_{15}	1	1	1	1	1	5
u_{16}	1	1	1	1	1	5
u_{17}	1	1	1	1	1	5
u_{18}	0	0	1	1	1	3

Summary:

The parameterization reduction has a powerful effect especially in large soft set. The following are the advantages and disadvantages of this algorithm:

Advantage:

i) Their optimal and sub extraction is correct.
ii) It provides consistency in selecting optimal and sub optimal decision.
iii) It reduces soft set size.

Disadvantage:

i) Parameter reduction is partially successful because of the inability to remove all the non-implies.
ii) It has no row reduction.
iii) It contains some implies inconsistency because deleting a single parameter can detect only some of the inconsistencies.
iv) The complexity of sub set combinations in some cases is not enhanced based on single columns.

2.7 Soft Set Comparisons

Several techniques have been utilised for soft parameter reduction using soft-set theory, which has contributed to the field of data reduction. The approach using soft-set reduction, first introduced by Maji et al. (2002), was implemented to determine decisions based on AND, OR, NOT, union and intersection. Parameterization reduction was then presented by Chen et al. (2005), who utilized rough set theory to investigate whether or not Maji's decisions were correct. Chen et al. successfully reduced the inconsistency as encountered by Maji et al., however, they inherited the problem related to sub-optimal choice faced by Maji et al. This problem was then analyzed by Kong et al. (2008), who defined the actual parameter reduction to overcome the sub-optimal problems as well as reduce the number of parameters; however, their algorithm was difficult to understand. Thus, Rose et al. (2010) presented a framework in decision making that solved the complexity of Kong's algorithm by using easier definitions. Later, Mamat et al. (2011) enhanced Kong's normal parameter reduction by easy definitions using different steps than that of Rose et al. (2010). Then, Rose addressed the idea of raw reduction (Rose et al., 2012) that enhanced the algorithm that Rose et al. (2010) suggested by first reducing the implies before putting every object in the reduction set if they are maximally or minimally supported by parameters. After this process, if there is zero significant parameter, it will also be under reduction. Kumar and Rengasamy, (2013) introduced the idea of deleting every parameter and matching its remaining parameter partitions with the original partition; if they are the same then the deleted parameter goes for reduction. The reduction from previous algorithms in Table 2.2 are shown in Table 2.7 and the reduction is summarized in Table 2.7. The limitation of the above reduction algorithms, as in Table 2.7, shows that in some cases less reduction was performed than in others, which is our motivation to achieve successful reduction by enhancing choices cost and decision quality as well as simplifying the decisions by reducing its boundary to fill the gap as represented in Figure 2.5. This shows that the problems exist in every techniques put forward by previous researchers (Maji et al. (2002); Chen et al. (2005); Kong et al. (2008); Mamat et al. (2011); Rose et al. (2010); Rose et al. (2012); Kumar and Rengasamy (2013).

Table 2.7 The reduction of previous soft parameters algorithms

Features / Reduction techniques	Raw reduction	Consistency (Correct optimal and sub optimal decisions)	Total reductions of Table 2.2	Problems
Maji 2002			33%	Inconsistency
Chen 2005			50%	Sub optimal decisions
Kong 2008		√	0%	Non- implies
Rose 2010		√	0%	Non- implies
Rose 2012	√	√	37%	Cases of non -implies
Kumar 2013		√	17%	Cases of implies

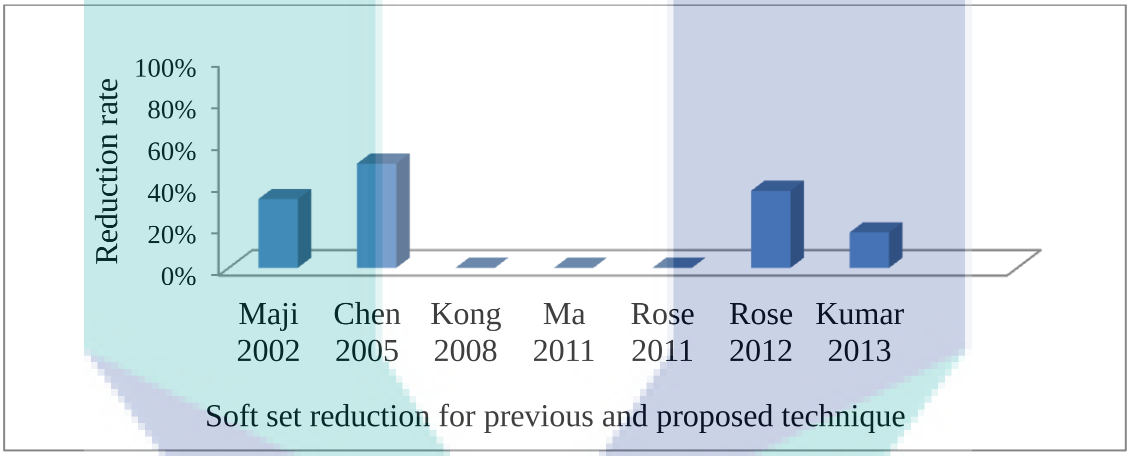

Figure 2.5 Reduction comparisons based on previous algorithms

2.8 The Benefit of Parameter Reduction

The benefit of parameter reduction is that it simplifies the process by enhancing the searching time and generating decisions without incurring any cost. The decompositions introduced by Ibrir and Bettayeb (2015) is to define an equal spacing between classifications. Decision analysis, first studied by Maji et al. (2002) and then by Chen et al. (2005), used to find optimal solutions that yield fewer number of parameters is not a simple process, but if the number of parameters is not reduced, the cost of choices will be increased. The question is which characteristic should be assigned to a given boundary. For this reason soft-set defined the equal spaces between classification to reduce implies, as suggested by Kong et al. (2008), Rose et al. (2010), Ma et al. (2011) and Rose et al. (2012). However, a problem of not giving exact classifications is encountered if the spacing between preferences is not equal, but the problem can be overcome by searching the solutions in single sub set combination decomposition, as introduced by Kumar and Rengasamy (2013). The proposed SSR algorithm in this study does not require single sub sets combination since the parameters reduction were considered in solving the problem related to time series co-efficient (refer Table 2.8 in Appendix A1). Time series is defined as the total frequencies that occur in a specific time and their similarity determined based on lengths (Zolhavarieh et al., 2014), in which the grid K differences have an equal spacing. The objects in time series frequently changes, such as in time series $f_i = \{f_1, f_2, ...f_n\}$ that is an ordered set of numbers that indicate the temporal characteristics of objects at any time t of the total track life T. Data similarity between two time series is based on the similarity in each time step in order to match certain characteristics for the grid independent. This method was introduced in soft-set by Hakim (2014), which is used to analyze the characteristics of the items and gives the recommendations based on situations, if the situation occurs in the grid tested, then the consequences will be the result. Suppose in two soft-sets (F_1, E_1) and (F_2, E_2), the solutions is checked by the probability that "If (F_1, E_1) then (F_2, E_2) or if (F_2, E_2) then (F_1, E_1)".

On the other hand, the benefit of parameter reduction is to lessen the computational duration of candidate solutions to enhance their further reduction, and simultaneously maximize the pre-reduction stage to become more significant, which can optimizes the usage of CPU; in some cases, the transfer of resources is increased if data is not reduced. The purpose of the decomposition is to establish partial reduction within the allowable range of the implies; if the reduction achieved is small, then the overall boundary of the solutions is affected by the change. The computational time required to find the boundary is further reduced, and at the same time, overcoming some of the incorrect sets from the boundary of candidate solutions, while decreasing the capacity of the buffer.

2.9 Summary:

The reduction of objects or dimension (attributes) in a soft-set that contains some amount of redundancies could not assist in the discovery of knowledge. Thus, these redundancies need to be eliminated to allow only the relevant attributes to be used in the discovery of knowledge. In order to determine the reduction, soft-set theory and rough set theory have been used to manage uncertain data. Many years ago, the data set costs more and consumed large memory size due to the presence of uncertain data as unlimited resources need to be allocated to the set. If the uncertain data is reduced, then it would improve the quality of information as well as the choices cost. The purpose of reduction in decision making is to identify the significant relevant data to make the cost of choices accurate and to maintain consistency in attaining the optimal and sub-optimal choices.

Several techniques can be used for parameter reduction by dealing with uncertainty in soft-set theory to achieve reduction based on information characteristics. However, although the soft-set reduction is successful in reducing implies and some non-implies condition, it failed to provide consistency in both implies and non-implies. The study on soft-set reduction by Mohammed et al. (2014) has compared the effect of several soft attribute reduction techniques forwarded by previous researchers such as Maji et al. (2002), Chen et al. (2005), Kong et al. (2008), Mamat et al. (2011), Rose et al. (2010), Rose et al. (2012) and Kumar and Rengasamy (2013). First, in Maji's algorithm, the optimal and sub-optimal decisions generated based on maximum weight has many inconsistencies and incorrect decisions. The inconsistency was removed by Chen et al., but Chen et al.'s sub-optimal result is also incorrect. The performance of sub decisions in Chen's algorithm was improved by Kong et al. by introducing implies conditions, in which the optimal and sub-optimal accuracy were maintained. However, in some cases, if implies have not taken place, sometimes only partial reduction will take place because it excludes non-implies. Rose et al. (2010) introduced simpler definitions that capture implies as in Kong et al.'s algorithm; then, by using the easy definition, Mamat et al. enhanced Kong et al.'s complexity. In Rose et al.'s algorithm, the concept of reducing non-implies was introduced along with implies definitions (Rose et al., 2010). Kumar and Rengasamy (2013) extended the concept of non-implies to be applied in each single parameter, but this concept does not consider some implies and even some non-implies as reduction; the problem they sometimes encountered was that some amount of vagueness cannot be filtered. Thus, if implies or non-implies are placed in more than one column, the algorithm cannot longer capture it. This current review is useful in enhancing characteristic of decision partition classifications that are not stable. However, these cases do not maintain the accuracy of the original classifications. Another issue is that the reduction efficiency is not enhanced if the domain of sub sets are not feasible due to large parameters. The large parameters induced infinite combinations and, for this reason, previous studies decomposed small sub sets by removing the complexity of feasible domain based on single column (Herawan et al., 2009 a; Mamat et al., 2011; Kumar and Rengasamy, 2013). However, removing single parameters does not always enhance the reduction efficiency, thus it needs more decompositions.

CHAPTER 3

METHODOLOGY

3.1 Introduction

This chapter discusses the research methodology that was conducted in this study in developing the proposed hybrid algorithm that is used for managing uncertain data and maintain the consistency during parameter reduction process. Afterwards, the mathematical model is presented to formulate the probability of parameter reduction based on adjusted prioritized weight vector and Markov Chain model. Then, the detailed development of the proposed efficient Soft-Set Reduction based Binary Particle Swarm optimized by Biogeography-Based Optimizer (SSR-BPSO-BBO) algorithm is discussed as a method to perform soft-set data classification and optimal decision making processes. Finally, a summary of the proposed algorithm is given.

3.2 The Definitions of Soft Set

In developing the proposed soft-set combinations technique, reference was made to several previous soft-set algorithms as FPC combination, such as those proposed by Rose et al. (2010) and Kumar and Rengasamy (2013). The definitions of theses algorithms are as follows, which are later used in the proposed combination to maintain original inconsistency:

Definition 3.2.1:

Let *(F, E)* be a representation of soft parameters as parameters and objects, and their values are determined by the universe *U*, where $u \in U$. A parameter co-occurrence transforming the object *u*, where their values arrangement can be defined as $coo(u) = \{e \in E : f(u,e) = 1\}$, refer (Rose et al., 2010) for details.

Definition 3.2.2:

Let *(F, E)* be a soft-set over the universe U and $x \in U$. Support of an object u rules can be expressed as $\operatorname{supp}(u) = \operatorname{card}(\{e \in E : f(u,e) = 1\})$ of $coo(u) = \{e \in E : f(u,e) = 1\}$, refer (Rose et al., 2010) for details.

Definition 3.2.3:

Let *(F, E)* represents soft parameters and their values are determined by the objects from the universe U and $u \in U$. An objects u could achieve maximally supported status to be ranked at first if it contains all parameters E, denoted by $\mathrm{Msupp(u)}$, if the source of the term $\mathrm{supp(u)} > \mathrm{supp(v)}$, $\forall v \in U/\{u\}$ to perform order (Rose et al. 2010). Based on Definition 3.2.3, the supported (ranked) ordered objects can be made according to the above form and is conducted in such a way that their support value are categorized in descending order as $U_1 > U_2 > .... > U_n$, in which their values can be obtained from $\mathrm{U_i} \subseteq U$ and $U_i = \{u \in U : u \text{ is } i\text{-th maximal supported by } E\}$, for $1 \le i \le n$. Thus, U_i is a collection of objects as neighboring nodes, such that for every I that belongs to U should have the same support; in other words, the various parameters representing the objects that has the same support should be placed in the same class). The ordering solutions is expressed as $U = \bigcup_{I<i<n} U_i$ and $U_i \cap U_j = \phi$, for $i \neq j$. In addressing the rows summation, a collection of $U/E = \{U_1, U_2, ..., U_n\}$ induced a decision partition of U, refer (Rose et al., 2010) for details.

Definition 3.2.4:

Let (F, E) be the representations of finite soft parameters and their variable coming from the universe U, where the partition of the original set is U/E. To determine the partition of original sub sets, let $c \in E$. Then c is the indispensable representations in E if $U/\{c\} = U/E$. Otherwise, c is said to be dispensable, refer (Kumar and Rengasamy, 2013) for details.

Definition 3.2.5:

Let (F,E) be the representations of objects in soft-set identified in the universe U, $C \subset E$ is the parameters sub set, and let a single parameter is $c \in C$. Then, parameters $\{c\}$ is dispensable if $U/\{c\} = U/E$, otherwise, $\{c\}$ is indispensable, which is used to avoid the up growth of parameters. To overcome all dispensable parameters that all ci values will be removed, refer (Kumar and Rengasamy, 2013) for details.

Definition 3.2.6:

Let $A \subset E$ define the uncertainty that A is dispensable in E if $U/A = U/E$, otherwise the unsatisfaction principles makes A as indispensable in E.

Definition 3.2.7:

Let $A \subset E$ has implies derivations that A is usually a reduction in E if and only if the correction of A in Definition 3.2.6 is dispensable and $\mathrm{supp}_{E \backslash A}(u) = \mathrm{supp}_{E \backslash A}(v)$, for every $u, v \in U$, (Rose et al., 2010). For example after removing a particular sub set their remains parameter may induced original set decision partition, for this reason the removable set must generate equality for all objects.

3.3 Proposed Soft Set Parameter Consistency Algorithm

HPC algorithm has been designed based on Hybrid Parameter Reduction (Rose et al., 2012) and Parameterization Reduction (Kumar and Rengasamy, 2013). The purpose of this combination is to overcome the limitations of both techniques. The HPC combinations reduce similar relations from objects based on new objects relations. The reduction based on Hybrid parameter reductions by Rose et al. (2012) removes the implies, then it deletes the objects that have similar representations of parameters, which will then remove every zero significant parameter. Thus, the issue of non implies is not solved using Rose's algorithm (Rose et al., 2012), except that the empty column induced by similar objects is removable. Hence, the proposed hybrid algorithm in this study would decreased the choices cost, which consequently affects storage and transformation, while searching for significant parameters, as shown Figure 3.1. The similarity between the two algorithms is that it reduces non implies using different strategies.

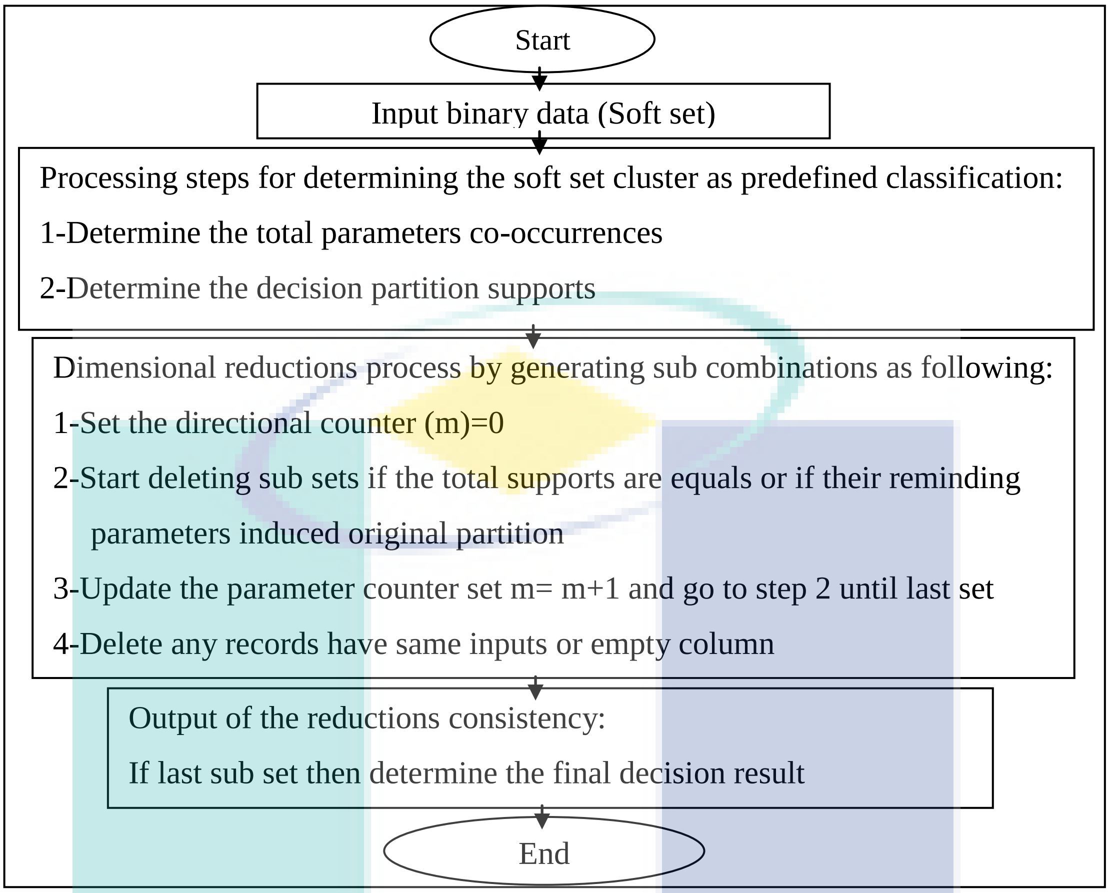

Figure 3.1 The process of the proposed HPC algorithm

Some non implies from finite parameters volume are shared by both algorithms from different aspects (Kumar and Rengasamy, 2013; Rose et al., 2012), which are induced by empty columns introduced from Rose et al. (2012). However, Kumar and Rengasamy's algorithm has automatic decompositions, based on singleton reduction for both choices cost and domain space reduction. However, the problems that arise in both algorithms is in terms of singleton sub set. Thus, to overcome the problems, the proposed combinations would be to improve Kumar's algorithm, and to reduce more sets using Rose's algorithms than that obtained from using algorithms put forward by previous researchers.

3.3.1 HPC Definitions

The following refers to the definitions cited in Rose et al. (2010):

The representations of soft-set in Definition 3.2.1 is used to identify Boolean information; Definition 3.2.2 is used to calculate the weight; Definition 3.2.3 is used to determine the clusters of the original set; Definitions 3.2.2 and Definitions 3.2.3 are used to determine the similarity of objects; Definition 3.2.4 is used to delete single parameters from Kumar's algorithm; and Definition 3.2.5 is used to overcome all deletions in Kumar's algorithm. The new soft set generated (based on Kumar's deletions and further reductions by Rose's (2010) technique is used in Definition 3.2.6 to determine the feasible implies boundary, then it is matched using Definition 3.2.7 to overcome the implies from it, and then similar objects are simplified using Definitions 3.3.1.1, after which all similar objects are deleted using Definitions 3.3.1.2. These definitions setup focuses on uncertain parameters for implies and non implies, which facilitates the reductions based on FPC combination introduced by Rose et al. (2010) and Kumar and Rengasamy (2013), while the proposed combination contributes a new Definition 3.3.1.3 for generating significant reduction based on AND condition.

Definition 3.3.1.1:

Let (F,E) represents a set of parameters mapping of given objects placed in the universe U and $d \in U$. Then, d is used to simplify the objects; the simplifications is called dispensable U if d is maximally or minimally supported by E, otherwise, d is said to be indispensable (Rose et al., 2012).

Definition 3.3.1.2:

Let (F,E) represents soft parameters expressed by the removable of set D in Definition 3.3.1.1, if there are empty parameters, then the parameters are deleted (Rose et al., 2012).

Definition 3.3.1.3:

Set A is forwarded as reduction if and only if the set satisfies the conditions: C is dispensable (Definition 3.2.5) and A is dispensable (Definition 3.2.7) and set d (Definition 3.3.1.1). As an example, set C contains the removable parameters obtained after implementing Kumar's algorithm (Kumar and Rengasamy, 2013), the implies are then removed by using set A (Rose et al., 2012), and finally, set D contains the objects that have similar values as in the original sets.

3.3.2 HPC Combination Procedure

The detailed procedure of the proposed HPC algorithm is explained as follows:

1- Filter parameters by Parameterization Reduction Using Soft-set Theory for better Decision Making.

2- Execute Hybrid reduction algorithm.

3.3.3 HPC Process

The procedure of the proposed HPC reduction combinations is as follows:

1- Accept original soft-set (F, E).
2- Determine the parameter co-occurrences in every object based on Step1.
3- Calculate the total of parameters supports in every object based on Step 2.
4- Determine object's decision partitions weight.
5- If any object is maximally or minimally supported by parameters, name as set *D*, where $D = \{D_1, D_2, ..., D_n\}$, and repeat until the last object.
6- In every parameter, calculate their complements objects support, if the values are the same as that of the original set, then Forward *C* for its all parameters, where $C = \{C_1, C_2, ...\}$.
7- Union the C columns and remove from the original set.
8- Based on Step 7, for any sub set $A \subset E$, if $U / A = \boldsymbol{U} / \boldsymbol{E}$ and $f_A(P_1) = f_A(P_2) = ... = f_A(P_n)$, then forward L.
9- Remove maximum cardinality of set *L*.
10- For any D_i, remove from set *E*.
11- If there is empty column, then delete it.
12- This algorithm known as HPC algorithm.

3.4 Object Reduction as Complete Sub Cluster Reduction

The proposed definitions will remove uncertain objects as proper reduction cannot be determined without false frequent object. By removing false frequent object using AND and OR operations, union of all intersection occur as a result of reduction. The objects are deleted if their parameters in union intersect. First, the reduction of object is determined based on hybrid reduction of the proposed method, and then Jaccard are placed in measuring the object reduction co-efficiency based on AND and OR operations. Let B be any sub set of U and let X be the reduction coming from parameters decision support.

3.4.1 Reduction prosperities

Definition 3.4:

Let any be two sub sets of E and that E is the original set. Suppose $A, \ldots\ldots \subset E,$ then for any $c \subset A, B, \ldots\ldots,$ the union over intersections of {c}, based on Jaccard similarity, should be the same (Jaccard, 1902).

The basic steps of the proposed algorithm are as follows:

Step 1:

Let co-occurrences induced decision partition $c = \{u_1, u_2\},$ suppose $E = \{u_1, u_2\},$ $A = \{u_1, u_2\},$ by this example, Jaccard similarity is calculated as in Step 2.

Step 2:

The intersection between two sets (A, E), from the example above, is determined such that $\frac{A \cup E}{A \cap E} = 1.$ Suppose set X = $\{A, B, \ldots\ldots.\}$ and $\{A, B, \ldots\ldots.\} \subset E,$ then all {c} is known as the first intersection, and the intersection continues until the last result is obtained. If there is any intersection, determine it. This step determines the intersection between all objects as first intersect, then it intersects until the last intersection $E = \{Min, \ldots\ldots, Max\}$ as Max= $\overline{B}(X)$ and Min= $\underline{B}(X)$.

Step 3:

Take sub sets that induced the original partition from Step 1 as set B, and the values of the reduction result obtained from the algorithm, which retains the original table, are entered and the result is gated as {c}, then find $\underline{B}(X)$ as reduction.

3.5 Mathematical Model of Parameter Reduction for Decision Partition Order Based on Adjusted Weight Vector

The probability of occurrence of at least n suboptimal choices in a given soft-set can help us to investigate the key factors which have impact in suboptimal choice redundancy. In this regard, a suboptimal choice of subset P is considered as shown in Figure 3.2, where in each choice combination in the subset P has a number of repetition and priority value.

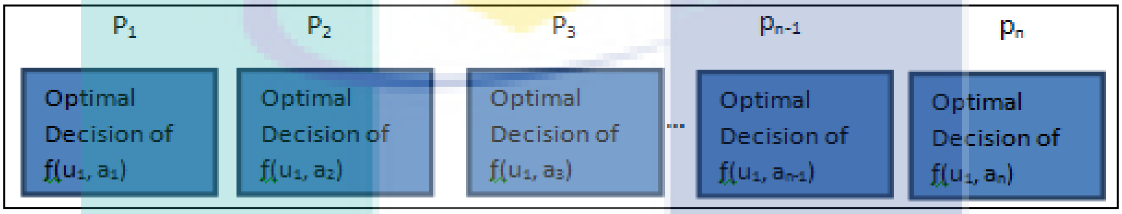

Figure 3.2 Probability representation of having an optimal decision within sub-set parameter of f_n.

There are N sets of optimal decisions in a soft-set and each combination of f (u, a) has variables or features, fi, and priority of reduction rate Pr. Optimal decisions are exponentially distributed in the soft-set with parameter lambda which refers to the mean value of Pi. Based on our assumptions, sub-sets are partially equipped with weighted priority which affects the optimal decision (Od) classified and ranked according to the value of below equation based on a proposed weight vector, W is identified to present the weight of Od.

$$\overline{W} = W_{f(u_1,a_1)}, W_{f(u_1,a_2)}, W_{f(u_1,a_3)}, \dots, W_{f(u_1,a_{n-1})}, W_{f(u_1,a_n)} \tag{3.1}$$

In order to obtain an optimal weight vector, an adjustment value is calculated with respect of standard deviation of each combination of suboptimal choice as presented by the following equation:

$$\bar{A} = \left(A_{f(u_1,a_1)}, A_{f(u_1,a_2)}, \dots, A_{f(u_1,a_n)}\right) = \left(\frac{\sigma_{f(u_1,a_1)}}{_{i}\sigma_i}, \frac{\sigma_{f(u_1,a_2)}}{_{i}\sigma_i}, \dots, \frac{\sigma_{f(u_1,a_n)}}{_{i}\sigma_i}\right)$$

$$i\epsilon \{f(u_1,a_1), \dots, f(u_1,a_n)\} \tag{3.2}$$

For instance, when the first sub-set P_1 of Od is considered to be given high priority based on the number of repetitions, the given weight vector equation will be as follows:

$$\bar{W} = \frac{A_{f(u_1,a_1)}}{M_{f(u_1,a_1)}} \times M_{f(u_1,a_1)} \times A_{f(u_1,a_2)} \times A_{f(u_1,a_n)} \tag{3.3}$$

The standard deviation of the sub-set P values have been normalized in above equation, where σi is the standard deviation of $P_{(i,1)}$, $P_{(i,2)}$, . . . , $P_{(i,n)}$ and M_i is their mean calculated utilizing the following equations:

$$M_i = \frac{1}{n}, n_{j=1} P_{i,j}, \qquad i\epsilon \{f(u_1,a_1), \dots, f(u_1,a_n)\} \tag{3.4}$$

$$\sigma_i = \overline{\frac{1}{n}, n_{J=1} (P_{i,j-}M_i)^2,} \qquad i\epsilon \{f(u_1,a_1), \dots, f(u_1,a_n)\} \tag{3.5}$$

Based on our proposed method, the sub-set P with high probability of reduction is considered a main input metric for decision making in the optimal decision election process; thus, it should be highlighted that low variance of sub-set P should not be reflected in a decrease in its own weight vector. For example, when the overall average of $f(u_1, a_j)$, (j = 1, 2, . . . , n) is with high total number of active options/ones, the adaptation degree of sub-set metric $f(u_1, a_j)$ must be considered with high priority to be ranked as first optimal decision. Therefore, the weight vector of- $f(u_1, a_j)$, $W\, f(u_1, a_j)$, should be given a higher weight value compared with the other optimal decision sub-sets. Accordingly, the reduction probability is significantly improved by guaranteeing that the optimal decision was made based on $f(u_1, a_j)$ with high priority weight vector.

3.5.1 Proposed Markov Model Based on Probability for Searching Strategy

Generally, the probability theory studies chance occurrence processes for which the knowledge of prior outcomes influence predictions for future experiments. In another words, when we observe a sequence of unplanned experiments, all of the previous outcomes could affect our predictions for the subsequent experiment. For instance, this would be the case in forecasting flood rates on a sequence of water levels in a region. However, to enable sequential process would most probably make it very tough to prove general results. We can define a Markov chain as follows: We have a set of states, $S_{f}=\{S_{f0}, S_{f1}, S_{f2}, \ldots, S_{fn}\}$. The process starts in one of these states and moves successively from one state to another. Each move is called a step. When the chain is currently in state S_{fi}, it then moves to state S_{fj} at the next step with a probability denoted by P_{ij}, and this probability does not depend upon which states the chain was in before the current state.

The parameter partition reduction probability of each parameter depends on the order of each partition in the sub-set. Hence, based on the order properties of parameters in the sub-set, it is likely that at least one or more sub-sets will be reduced in each cycle as a way to obtain an optimal decision. In order to compute the probability of at least x reduced sub-set in a cycle, which is known as a tail probability of x reduction, which is shown with $P_r(x,n,m)$ where n refers to the number of parameters with one sub-set and m is the possible number of combinations. Thus, we have suggested an approach based on Markov modelling method as it's shown in Figure 3.3. The proposed Markov model has N states in which each state fi refers to at least i candidate parameter in the sub-set be omitted from the new generated optimal decision set and all transitions start from f_0. Figure 3.3 presents the proposed Markov Chain model that demonstrates the probability concept of our parameterization value reduction technique (Norris and James, 1998.). It is worth noting that our proposed technique relies on three main phases. The first two phases include: ranking the sub-sets f_n based on the number of parameters offered by each one (accumulative number of ones in a row), and ranking each parameter by the number of times it is present with each f_n (accumulative number of ones in a column). The third phase is to guarantee an optimal decision is achieved by applying our proposed priority based weight vector.

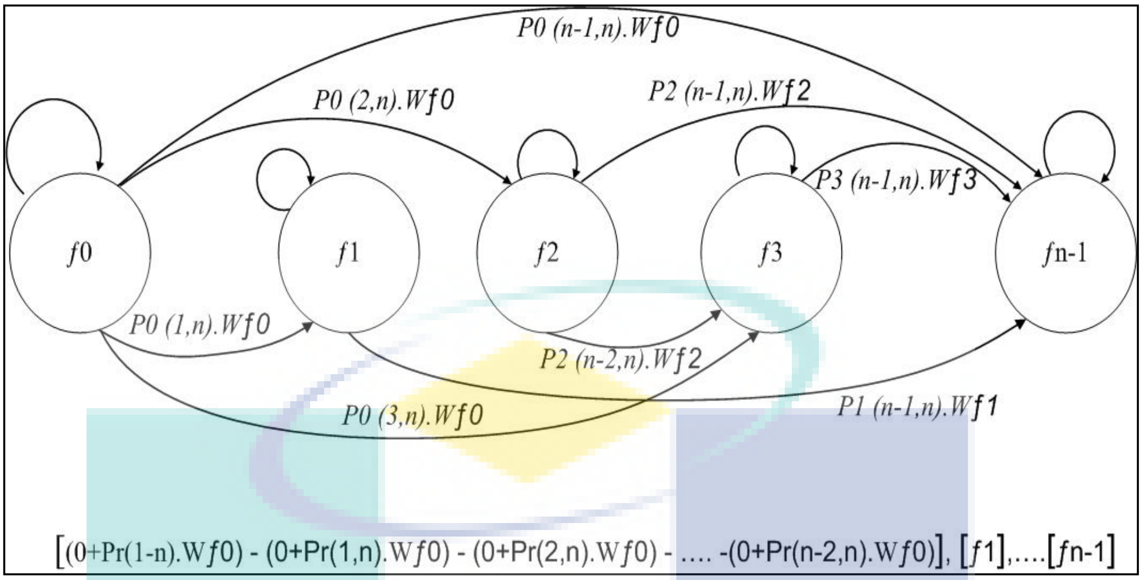

Figure 3.3 Representation of the proposed Markov Chain model

For a better understanding, an example is given of the way to calculate the value of tail probability, when at least one sub-set is selected to be reduced from the list of combinations.

$$P_r(1{,}4) = P_{f0} - P_{f1}(0 + P_{f0}) - P_{f2}(0 + P_{f0})(0 + P_{f1}) - P_{f3}(0 + P_{f0})(0 + P_{f1})(0 + P_{f2}) \quad (3.7)$$

P_r indicates the probability of reduction in the rate of fi in the sub-set which is explained in the following:

In the aforementioned equation, the calculation of the tail probability that two parameters are reduced in a sub-set consisting of five parameters. All possible combinations that could be used in selecting two options as an optimal decision out of five parameters should be considered. According to this theory, to calculate the tail probability of two reduced parameters in a sub-set, the tail probability of one parameter to be selected as the optimal decision should also be taken into account. Thus, transition between two consecutive states f_{i+1} and f_i where i >= 1 is not possible and transition probabilities and state probabilities are the same. The transition probabilities for at least one to three reduced parameters in N parameters length of a sub-set is shown as follows:

$$P_r(1,N) = \sum_{i=1}^{N-1}\left[\times \prod_{m=0}^{i-1}(0 + P_{fm}) \times W_{fm}\right] \quad (3.8)$$

$$P_r(2,N) = \sum_{i=1}^{N-2}\left[P_{fi} \times \left(\prod_{m=1}^{i-1}(0 + P_{fm}) \times W_{fm}\right) \times \left(\sum_{x=i+1}^{N-2}\left[P_{fx} \times \prod_{n=j+1}^{x-1}(0 + P_{fn}) \times W_{fn}\right]\right)\right] \quad (3.9)$$

$$P_r(3,N) = \sum_{i=1}^{N-3}\left[P_{fi} \times \left(\prod_{m=1}^{i-1}(0 + P_{fm}) \times W_{fm}\right) \times \left(\sum_{x=i+1}^{N-2}\left[P_{fx} \times \prod_{n=j+1}^{x-1}(0 + P_{fn}) \times W_{fn}\right]\right) \times \left(\sum_{y=n+1}^{N-1}\left[P_{fy} \times \prod_{v=y+1}^{y-1}(0 + P_{fv}) \times W_{fv}\right]\right)\right] \quad (3.10)$$

Finally, we introduced a general recursive function for computing the tail probability of *nr* number of reductions in a sub-set which is represented by Equation below:

$$P_r(f0,nr,N) = \left\{\sum_{i=f0}^{N-nc}\left[P_{fi} \times \left(\prod_{j=f0}^{i-1}(0 + P_{fj}) \times W_{fm}\right) \times P(i+1, nr-1, N)\right], nr \neq 0\right. \quad (3.11)$$

where, f_0 refers to the first parameter of a sub-set and N indicates the number of parameters given by a sub-set. W_{fm} indicates the calculated weight value of a given parameter (refer to Section 3.5, P. 62).

3.5.2 Markov Model Dimensional Reduction Algorithm

The algorithm of Markov chain steps as following:

1- Input original soft-set as set *M*.

2- In every object, determine the objects support partition and label as set *P*. Then decompose set *P* into set *Min* is Min set, *Max* is Max supp and set *PP* is R the remains.

3- Arrange Step 2 in descending order or ascending order, and label as set *Q*.

4- From set *M*, generate partial combinations of set *M* as set *W* components; search for the solution by checking the combination of set *W* in Step 6 to determine whether or not the components match the decision partition.

5- If last sub set, then go to Step 12, else add set $D_{(p+1)}$ to set $S_{(p+1)}$; update set W_i by maximum set $S_{(p+1)}$, and remove it if matches set *Q*; for new W_i go to Step 6.

6- For set W_i, if the object support partition of any W_{pi} induced a partition matches that of set *Min*, then forward to set *S*.

7- In set *S*, if the object support partition of any S_i induced a partition that matches that of set *Max*, then forward to set *SS*.

8- In set *SS*, if the object support partition of any SS_i induced a partition matches that of set *PP*, then forward to set *BB*.

9- In set *BB*, if the order of object support partition of any BB_i induced a partition matches that of set *Q*, then forward to set *K*.

10- For current K_i, if the order of the decision partition matches that of set *Q*, then go to Step 11; else, if the decision partition order does not match that of set *Q* and if its removal makes the order of the decision partition of the remaining columns the same as that of set *Q* (by checking steps 7, 8 and 9), then forward set K_i to set $D_{(p+1)}$ and go to Step 5.

11- Display the columns of K_i as results, and go to Step 13.

12- Display the maximum columns not in set $S_{(p+1)}$ and the decision partition order of W_i that is the same as set *Q*, then go to Step 13.

13- End.

3.6 Accuracy of Original Decision Partition Order

Classification methods that are specialized to solve a specific problem can often achieve better performance in terms of accuracy and complexity time, which could be achieved by considering several features in addition to obtaining the background knowledge. In order to optimize the proposed Soft-Set-Reduction (SSR) algorithm, a new heuristic learning algorithm based on binary version of PSO (BPSO) is used to classify and reduce the low ranked parameters in a given soft-set. The binary version of this algorithm has been introduced for solving binary issues, specifically the visibility of parameters within sub-sets of a universal soft-set that are represented by 0s and 1s.

3.7 Proposed Soft Set Reduction using Binary Particle Swarm Optimization Based on Biogeography-Based Optimizer Algorithm (SSR-BPSO-BBO) to Predict the Accuracy of Decision Partition

The BPSO was proposed by Kennedy and Eberhartin (1997). Basically, the continuous and binary versions of PSO can be illustrated using two different components: a new transfer function and an altered technique for position updating process. For clarification, the process of mapping a continuous search space in a given soft-set to a binary one; a transfer function was used. On the other hand, the updating process is intended to switch positions of particles between 0 and1 in binary search spaces.

In order to obtain an optimized classification method that could be integrated into Soft Set Reduction (SSR), Back Propagation (BP) of Multi-Layer Perceptron (MLP) Neural Networks (NNs) were used in this study. The new SSR using BPSO that is trained by Biogeography-Based Optimization (BBO) SSR-BPSO-BBO algorithm could efficiently assist in obtaining the optimal decision for the give problem scenario. The general mechanism which is used in generating the final decision of selected parameters of given sub-sets is designed to be BPSO. In order to modify particles position during searching process, in our proposed SSR-BPSO-BBO algorithm each particle within PSO should consider its current position, the present velocity, the distance to their personal best solution, pbest, and the distance to the global best solution, gbest. The mathematical model of PSO is presented as follows (Kennedy and Eberhartin 1997):

$$v_i^{t+1} = wv_i^t + c_1 \times rand \times (pbest_i - x_i^t) + c_2 \times rand \times (gbest - x_i^t) \quad (3.12)$$

$$x_i^{t+1} = x_i^t + v_i^{t+1} \quad (3.13)$$

where, v_i^t is the velocity of particle *i* at iteration *t*, w is a weighting function that was suggested earlier, c_j is an acceleration coefficient, rand is a random number between 0 and 1, x_i^t is the current position of particle *i* at iteration *t*, $pbest_i$ is indicating the best solution that an *i-th* particle has obtained until this point, and gbest designates the best solution the swarm has achieved so far.

A random particle distribution in a problem space will be performed at the early stage of running a PSO. Afterwards, the velocities of particles are calculated using Eq. (3.11) and keep on updating during every iteration. When defining the velocities, the position of particles will be calculated using Eq. (3.13). The process of changing particles' positions will continue until satisfying an end criterion.

Commonly, there are many problems that have inherent distinct binary search spaces, like feature selection and dimensionality reduction (Mirjalili and Andrew, 2013). Besides, issues with non-stop real-time search space can be converted into binary problems by converting their variables to binary variables, which is related to the behaviour of soft-set that normally consist of combinations of 0 and 1.

The main concept behind distinct binary searching spaces, the position updating process lays on switching between 0 and 1 values. Thus, this type of switching or position updating should be done based on the velocities of particles. Based on (Mirjalili and Hashim, 2012), the concept of updating particles' velocity is introduced by applying a probabilistic model. The key idea is to change the position of any particle in a binary searching space with the probability of its velocity. Hence, to achieve this, a transfer function is necessary to change velocity values to probability values for updating the particle's positions. It is important to note that the utilized number of particles in our algorithm was 40, the learning rate 0.01 and the momentum 0.001.

$$\mathrm{T}\left(v_i^k(\mathrm{t})\right) = \frac{1}{1 + e^{-v_i^k(t)}} \quad (3.14)$$

The velocity of particle i at iteration t in *k-th* dimension is $v_i^k(\mathrm{t})$. By changing velocities to probability values, position vectors could be updated with the probability of their velocities as follows:

$$x_i^{\mathrm{k}}(\mathrm{t}+1) = \begin{cases} 0 & if\ rand\ < T\ (v_i^k(t+1) \\ 1 & if\ rand\ \geq T\ (v_i^k(t+1) \end{cases} \tag{3.15}$$

Figure 3.4 shows the steps of BPSO process in finding the optimal solution in a given searching space of soft-set parameter reduction. At the same time our proposed SSR-BBO algorithm works in training the BP of MLP in obtaining the lowest Mean Square Error (MSE) of predicted output. Figure 3.5 illustrates the overall proposed SSR-BPSO-BBO flow activities. The elements with optimal MSE value will be used in updating the final position vectors of all particles in a searching space, which contributes to obtaining an optimal decision with high percentage of reduction of given soft-set.

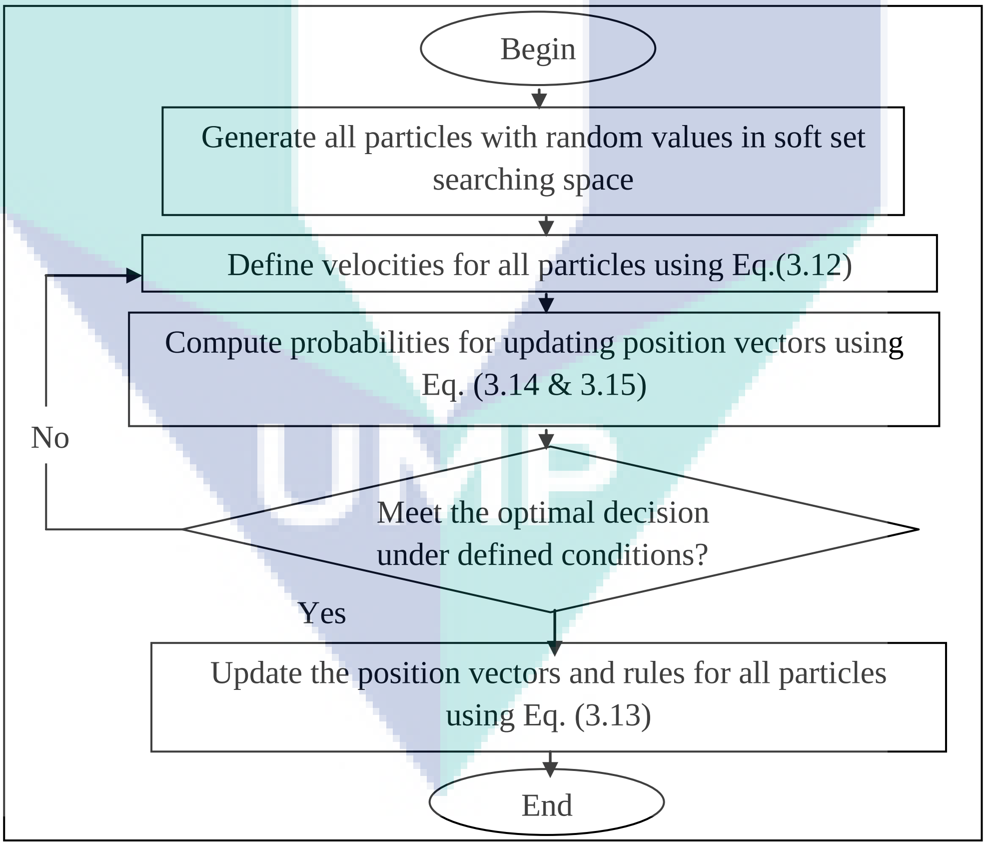

Figure 3.4 The flow chart of BPSO process

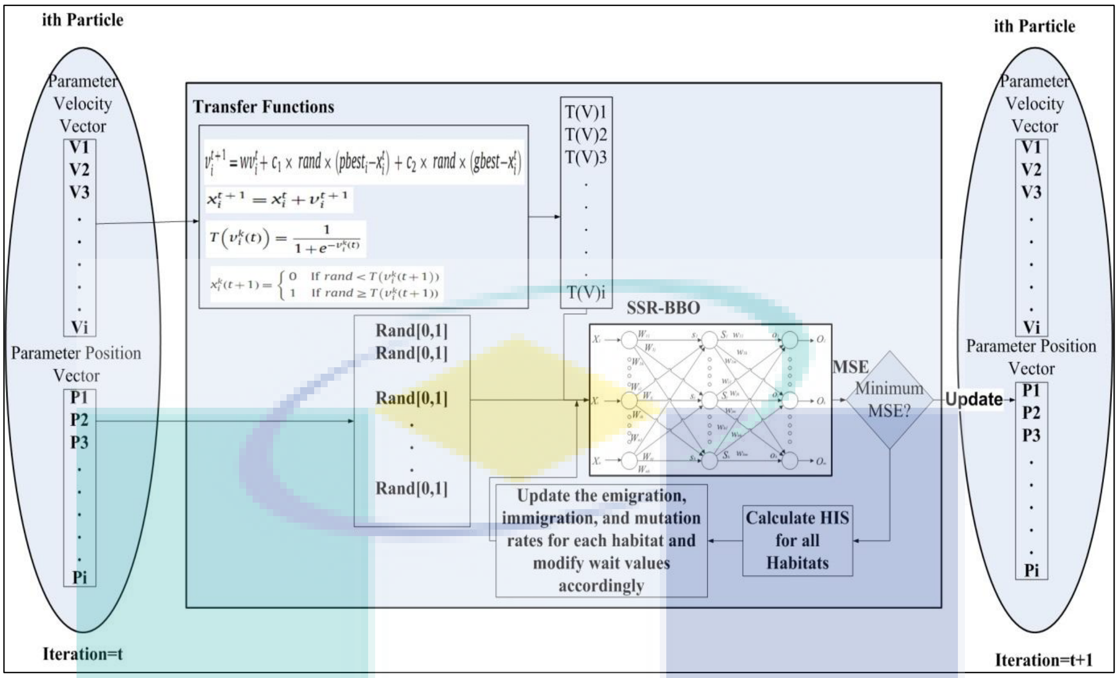

Figure 3.5 Proposed hybrid SSR-BPSO-BBO algorithm

In our proposed algorithm, the BBO sub-algorithm is applied to an MLP using the first method. Generally, the main aim of using BBO is train an MLP so that it is able to recognize training, validation, and test soft-sets completely for our use in parameter reduction process. The most significant sub-set in the learning phase is the training sub-set. The MSE function that was utilised in this study for all training samples is presented as follows (Mirjalili et al., 2014):

$$E = \sum_{k=1}^{n} \frac{\sum_{i=1}^{p} (o_i^l - Opt_i^l)^2}{n} \tag{3.16}$$

where, n is the number of training samples, p is the number of outputs, Opt_i^l is the optimal output of the ith input unit when the kth training sample is used, and o_i^l is the actual output of the ith input unit when the kth training sample appears in the input parameters.

3.8 Summary

This chapter presents and discusses the proposed algorithm. The Soft-Set Reduction algorithm consists of three sub-algorithms that are based on two components, namely, minimizing of domain space and adjusting the original classifications. HPC algorithm includes parameterization reduction that has quick decision for the reduction process. The HPC algorithm was created to remove uncertainty and vagueness from soft-set. This chapter also explains the methodology involved in developing the algorithm to increase the reduction efficiency along with decision accuracy. The proposed algorithm has the ability to improve the response time when dealing with large data set. To achieve an accurate decision partition order, the proposed algorithm filters the false frequent parameter reduction in soft-set theory.

In this chapter, a mathematical model that could perform parameter reduction based on adjusted weight vector and Markov Chain model has been detailed, showing the model's robustness in obtaining accurate optimal decision. Furthermore, an efficient Soft-Set Reduction based Binary Particle Swarm optimized by Biogeography-Based Optimizer (SSR-BPSO-BBO) algorithm which can be used in soft-set data classification and optimal decision making processes is discussed. The results obtained from the proposed model will be presented in the next Chapter.

CHAPTER 4

RESULTS AND DISCUSSION

4.1 Introduction

This chapter presents the results obtained from the proposed SSR and HPC algorithms and a benchmarking is implemented to validate our algorithms. The ability of the algorithms to find the optimal decision is presented and discussed by illustrating the reduction capability in a given soft dataset. Moreover, this chapter presents and validates the ability of the proposed algorithm in removing unwanted or less priority values inside the approximation range of a given soft dataset that has vagueness. Besides, the probability of reduction is demonstrated and analyzed using the proposed Markov chain model, while the average error in predicting the optimal decision is presented and compared with other optimization methods. Besides, the obtained optimal decision out of eight sub-sets and 200 parameters using the proposed SSR-BPSO-BBO algorithm is demonstrated and analyzed. Finally, the complexity in obtaining the optimal decision out of a given soft dataset of the proposed algorithm is investigated and benchmarked with existing methods.

4.2 Analysis of HPC Combination

The idea behind the proposed algorithm is to reduce the binary data; for example, suppose there is an original table named set E that contains rows (objects) and columns (parameters). HPC algorithm is used to determine parameters reduction of the set, which affects the identification of effective subsets. However, the total cost of determining and calculating the valuable data is affected by the formation of abnormal parameters subsets complexity. Besides other related issues such as decomposition complexity, classification and accuracy shortcoming, construction of subset is one of the major combinatorial problems that contribute to the development of soft-set reduction; major reduction problems have been dealt with using Rose et al. 's (2012) and Kumar et al. 's (2013) algorithms. Kumar et al. (2013) conducted a study to analyse the uncertainty cost by constructing new soft-set without removing any single parameter due to inadequate inconsistency interoperability among the soft set parameters. The study proved that the features of the remaining parameters will decide whether or not the removed parameter is significant. The algorithm deletes the parameters that do not induce the original cluster, leaving the remaining parameter clusters to be checked for the solution; the deleted parameters becomes part of parameter reduction. In the context of one or more columns removable by Rose's (2012), algorithm however, several non implies factors, including lack of consistency, are not considered in the reduction (Kumar and Rengasamy, 2013). Consequently, a technique has been developed in this research by combining Rose's (2012) and Kumar's (2013) techniques, which does not affect the rebuilding of the original table constructions. The deletion is based on single parameter and is significant if the rest of the parameters have the same characteristics as that of the original data; the original characteristics are summarized as decision partition. On the other hand, anomalous values that have the same cost for all statements are identified through Rose's (2012) algorithm that can be found in rows or columns, regardless of the number (Rose et al., 2012); the most important property of Rose's algorithm is that all of the values equal the total cost. The proposed HPC algorithm was implemented in Java program as Net Beans IDE, 8.0.2 and executed on Intel (R) Core (TM) 2 Duo CPU processer T6600@2.20 GZ with 3.00 GB RAM and running on 32-bit Windows 7 operating system.

4.2.1 Analysis of HPC Reduction Based on Table 2.2

The reduction process, made possible with the use of cluster and soft-set in arranging the subsets to extract and transform the information, is used to integrate the data to meet the requirements. Initially, the parameters are sorted in Table 2.2 as $\{p_1, p_2, p_3, p_4, p_5, p_6\}$, which produces 62 sets of possible combinations. The original table is classified into categories according to the objects' total cost as objects decision partition; the objects decision partition of Table 2.2 (refer to p. 37) is determined as U/E=$\{u_1, u_5, u_6, u_{12}, u_{14}\}$, $\{u_2, u_3, u_4, u_8, u_9, u_{10}, u_{13}, u_{15}, u_{16}, u_{17}\}$, $\{u_{11}, u_{18}\}$, $\{u_7\}$.

Rows are classified according to their values, and the values in each group must be the same. The groups, known as classification, are later divided into Min supp, R supp. The classification is used to match between the original set and the reduction set, such that its property must be identical to all the data that satisfy the conditions of Min sup. The groups are forwarded to feasible reduction and the reduction is confirmed based on the R sup, where the original Min sub set cluster is $Min_{\text{supp}} = U / R_{\text{supp}} = \{u_7\}$ and $R_{\text{supp}} = \{u_2, u_3, u_4, u_8, u_9, u_{10}, u_{13}, u_{15}, u_{16}, u_{17}\}, \{u_{11}, u_{18}\}, \{u_1, u_5, u_6, u_{12}, u_{14}\}$.

Table 2.2 Tabular representation of a soft set

U/E	p_1	p_2	p_3	p_4	p_5	p_6	$f(.)$
u_1	1	0	1	0	0	0	2
u_2	1	1	1	1	1	1	6
u_3	1	1	1	1	1	1	6
u_4	1	1	1	1	1	1	6
u_5	0	1	0	0	1	0	2
u_6	0	0	1	0	1	0	2
u_7	0	0	0	0	1	0	1
u_8	1	1	1	1	1	1	6
u_9	1	1	1	1	1	1	6
u_{10}	1	1	1	1	1	1	6
u_{11}	1	0	1	0	1	0	3
u_{12}	0	1	0	1	0	0	2
u_{13}	1	1	1	1	1	1	6
u_{14}	1	1	0	0	0	0	2
u_{15}	1	1	1	1	1	1	6
u_{16}	1	1	1	1	1	1	6
u_{17}	1	1	1	1	1	1	6
u_{18}	0	0	1	1	1	0	3

Initially, several techniques are used to adjust the actual soft-set classifications, from which a combination of two techniques was found to be able to eliminate the disadvantages of both techniques, while capitalizing on their strengths. The proposed HPC algorithm maintains optimal and sub-optimal choices to lessen the cost by sending smaller decision size through the transmissions lines, which will benefit the customers and decisions makers. The HPC combination has the ability to remove a lot of resources, which has not been explored using each technique present in the HPC combination, especially to detect the presence of and remove the non implies in more than one parameter resources; this is the main reason why each individual technique cannot detect the presence of non implies in more than one column. The combination of both techniques results in the reduction of the inconsistency.

The proposed HPC algorithm is implemented in two steps. On applying the algorithm to Table 2.2, the parameter p_6 is first removed from the table by using Kumar's (2013) algorithm, but the resultant set $\{p_1, p_2, p_3, p_4, p_5\}$, still preserved the original property of the objects decision partition. The removal of parameter p_6 shows that there is another property to be placed in uncertainty exposure. To check the possibilities whether or not any column can be deleted using Kumar's algorithm requires the knowledge that the properties of the rest of the columns are identical to that of the original data. If the properties are identical, then the column is deleted, and the process is repeated for the next column until the last column is reached. Thus, the application of the proposed HPC algorithm is first used to check the significance of each parameter using Kumar's algorithm to Table 2.2, which will delete any parameter if the properties of the remaining columns are identical to that of the original data.

The parameter sub set $\{ p_6 \}$ if deleted it introduces Min supp $\{u_7\}$ that is the same as the original soft-set Min supp, therefore, it considered as feasible reduction, to conformed it, the R_{supp} is $\{u_2, u_3, u_4, u_8, u_9, u_{10}, u_{13}, u_{15}, u_{16}, u_{17}\}, \{u_{11}, u_{18}\}, \{u_1, u_5, u_6, u_{12}, u_{14}\}$, as a result, the parameter $\{ p_6 \}$ in Table 4.1 of reduction was achieved using our proposed HPC algorithm up to 17% reduction.

Table 4.1 The Table 2.2 reduction based on HPC in first step by Kumar

U/P	p_1	p_2	p_3	p_4	p_5	$f(.)$
u_1	1	0	1	0	0	2
u_2	1	1	1	1	1	5
u_3	1	1	1	1	1	5
u_4	1	1	1	1	1	5
u_5	0	1	0	0	1	2
u_6	0	0	1	0	1	2
u_7	0	0	0	0	1	1
u_8	1	1	1	1	1	5
u_9	1	1	1	1	1	5
u_{10}	1	1	1	1	1	5
u_{11}	1	0	1	0	1	3
u_{12}	0	1	0	1	0	2

Table 4.1 Continued.

U/P	p_1	p_2	p_3	p_4	p_5	$f(.)$
u_{13}	1	1	1	1	1	5
u_{14}	1	1	0	0	0	2
u_{15}	1	1	1	1	1	5
u_{16}	1	1	1	1	1	5
u_{17}	1	1	1	1	1	5
u_{18}	0	0	1	1	1	3

In the second step of the HPC algorithm, the insignificant data in Table 2.2 is chosen by using Rose's (2012) algorithm based on decision partition of the table, which are classifiable at U/E= $\{u_1, u_5, u_6, u_{12}, u_{14}\}$, $\{u_2, u_3, u_4, u_8, u_9, u_{10}, u_{13}, u_{15}, u_{16}, u_{17}\}$, $\{u_{11}, u_{18}\}$, $\{u_7\}$. Sometime a table has a maximum sub sets, while in Table 2.2 some sets are not important because they are free from the property of the original data or it has deviation from the original property. Suppose that some of the sub sets in Table 2.2 are $\{\{p_2, p_3, p_4, p_5, p_6\}, \{p_1, p_3, p_4, p_5, p_6\}, \{p_1, p_2, p_4, p_5, p_6\}, \{p_1, p_2, p_3, p_5, p_6\}$, $\{p_1, p_2, p_3, p_4, p_6\}\}$. The partition of these sets is different from that of original soft-set, hence their deletion is not significant.

The equality property is determined by Rose's (2012) algorithm as implies conditions, from which 25 sets of the Min supp complement are the same as that of the original Min supp partition, such as $\{\{p_1, p_2\}$, $\{p_1, p_3\}$, $\{p_1, p_4\}$, $\{p_1, p_5\}$, $\{p_2, p_3\}$, $\{p_2, p_4\}$, $\{p_2, p_5\}$, $\{p_3, p_4\}$, $\{p_3, p_5\}$, $\{p_4, p_5\}$, $\{p_1, p_2, p_3\}$, $\{p_1, p_2, p_4\}$, $\{p_1, p_2, p_5\}$, $\{p_1, p_3, p_4\}$, $\{p_1, p_3, p_5\}$, $\{p_1, p_4, p_5\}$, $\{p_2, p_3, p_4\}$, $\{p_2, p_3, p_5\}$, $\{p_2, p_4, p_5\}$, $\{p_3, p_4, p_5\}$, $\{p_1, p_2, p_3, p_4\}$, $\{p_1, p_2, p_3, p_5\}$, $\{p_1, p_2, p_4, p_5\}$, $\{p_1, p_3, p_4, p_5\}$, $\{p_2, p_3, p_4, p_5\}\}$ and among them there is no set in which their complement satisfy the original set objects decision partition and the weight of the rows are the same. Thus, it implies that the choice value of each object is not changed after deleting parameter p_i. To assign the implies property to Table 2.2, 10 rows were reduced that are maximally supported by parameters $u_2, u_3, u_4, u_8, u_9, u_{10}$, $u_{13}, u_{15}, u_{16}, u_{17}$ which are found in parameters $\{p_1, p_2, p_3, p_4, p_5\}$.

In Step 2 of the proposed HPC algorithm, the reduction achieved is measured by using Rose's (2012) algorithm to define implies equality that reflect the data which are similar. These data are removed from the base table because they are contradictory, and we conclude that the similarity property tested over the rows and columns to generate the result. In this step, the Rose's (2012) algorithm generated the result as shown in Table 4.1, in which the sub-sets $u_2, u_3, u_4, u_8, u_9, u_{10}, u_{13}, u_{15}, u_{16}, u_{17}$ were deleted since the sub sets contain maximum ultimate objects support as shown in Table 4.2. However, there are neither complement in the parameters nor there are zero significant parameters in Table 4.2, from which the reduction based on HPC combination is found to be 37%. Using Rose's (2012) algorithm to check implies in Table 4.1, it can be concluded that implies is not found in the parameters. Hence, Rose's (2012) algorithm is used to check the rows of Table 2.2 to reduce the rows that are available at rows $\{u_2, u_3, u_4, u_8, u_9, u_{10}, u_{13}, u_{15}, u_{16}, u_{17}\}$, which are removable as shown in Table 4.2. Since implies property using for equality, suppose that E is the original set (such as Table 2.2), and suppose set A is any sub set where $A \subset E$, as in Table 4.2, the implies condition is satisfied if U / E = U / E-A and $\mathrm{supp}_{E \setminus A}(u) = \mathrm{supp}_{E \setminus A}(v)$, for every $u, v \in U$, where $u \subset A, E$. However, minimum ultimate objects supports do not occur in Table 2.2.

Table 4.2 The reduction in hybrid based on Table 2.2

U/P	p_1	p_2	p_3	p_4	p_5	p_6	$f(.)$
u_1	1	0	1	0	0	0	2
u_5	0	1	0	0	1	0	2
u_6	0	0	1	0	1	0	2
u_7	0	0	0	0	1	0	1
u_{11}	1	0	1	0	1	0	3
u_{12}	0	1	0	1	0	0	2
u_{14}	1	1	0	0	0	0	2
u_{18}	0	0	1	1	1	0	3

4.2.2 Analysis of HPC Reduction Based on Table 4.3

In the original data, certain features that summarize the characteristics of the data can be extracted, and the features are also shared by other data. The process of feature extraction is called classification, and each generated characteristic is unique.

When a particular data is found to be a reduction in accordance with the given property, the data is set to reduction as long as the original data does not spin into uncertainty.

The proposed HPC combination algorithm is then applied to Table 4.3 to determine the reductions. The reduction of the table based on the algorithm is presented and discussed in this section to evaluate the performance of the proposed HPC algorithm. Execution of HPC algorithm is the same as that in Section 4.2.1, but on a different table. In the process of reducing Table 4.3, the cluster partitions of the original soft-set are as follows:

The partitions = $\{\{u_1, u_{13}, u_{14}, u_{23}, u_{24}, u_{26}, u_{28}, u_{30}\}, \{u_4, u_7, u_{11}, u_{12}, u_{19}\}, \{u_5, u_6, u_8, u_9, u_{10}, u_{17}, u_{20}, u_{21}, u_{22}, u_{25}, u_{27}, u_{29}\}\}$. The Min supp cluster is $\{u_5, u_6, u_8, u_9, u_{10}, u_{17}, u_{20}, u_{21}, u_{22}, u_{25}, u_{27}, u_{29}\}$ and R supp cluster is $\{\{u_1, u_{13}, u_{14}, u_{23}, u_{24}, u_{26}, u_{28}, u_{30}\}, \{u_2, u_3, u_{15}, u_{16}, u_{18}\}, \{u_4, u_7, u_{11}, u_{12}, u_{19}\}\}$.

Table 4.3 Representation of framework for decision making soft set

U/P	p_1	p_2	p_3	p_4	p_5	p_6	p_7	$f(.)$
u_1	1	0	0	0	1	0	1	3
u_2	1	1	1	1	0	0	1	5
u_3	1	1	1	1	0	0	1	5
u_4	1	0	1	0	1	1	0	4
u_5	1	0	0	0	0	0	1	2
u_6	1	0	0	0	0	0	1	2
u_7	1	0	1	0	1	1	0	4
u_8	1	0	0	0	0	0	1	2
u_9	1	0	0	0	0	0	1	2
u_{10}	1	0	0	0	0	0	1	2
u_{11}	1	1	1	0	0	1	0	4
u_{12}	1	0	1	0	1	1	0	4
u_{13}	1	0	0	0	1	0	1	3
u_{14}	1	0	0	0	1	0	1	3
u_{15}	1	1	1	1	0	0	1	5
u_{16}	1	1	1	1	0	0	1	5
u_{17}	1	0	0	0	0	0	1	2
u_{18}	1	1	1	1	0	0	1	5

Table 4.3 Continued.

U/P	p_1	p_2	p_3	p_4	p_5	p_6	p_7	$f(.)$
u_{19}	1	0	1	0	1	1	0	4
u_{20}	1	0	0	0	0	0	1	2
u_{21}	1	0	0	0	0	0	1	2
u_{22}	1	0	0	0	0	0	1	2
u_{23}	0	1	1	0	0	1	0	3
u_{24}	0	1	1	0	0	1	0	3
u_{25}	1	0	0	0	0	0	1	2
u_{26}	0	1	1	0	0	1	0	3
u_{27}	1	0	0	0	0	0	1	2
u_{28}	0	1	1	0	0	1	0	3
u_{29}	1	0	0	0	0	0	1	2
u_{30}	0	1	1	0	0	1	0	3

Source: Rose et al. (2010).

From Table 4.3, based on the proposed HPC algorithm, its first step is to apply Kumar's algorithm (Kumar and Rengasamy, 2013), which is to execute parameterization process, but no reduction (0%) is achieved. Hence, the second step is to execute the hybrid reduction algorithms. Based on the HPC algorithm, the second step is to apply Rose's (2012) algorithm, in which p_6 and p_7 are removed because both parameters satisfy the two properties. After removing p_6 and p_7, the resultant cluster partition from the proposed HPC algorithm is the same as that of the original soft-set. The final reduction based on HPC algorithm as shown in Table 4.4 is 29%.

Table 4.4 Representation of Table 4.3 reduction based on HPC algorithm

U/P	p_1	p_2	p_3	p_4	p_5	$f(.)$
u_1	1	0	0	0	1	2
u_2	1	1	1	1	0	4
u_3	1	1	1	1	0	4
u_4	1	0	1	0	1	3
u_5	1	0	0	0	0	1
u_6	1	0	0	0	0	1
u_7	1	0	1	0	1	3

Table 4.4 Continued.

U/P	p_1	p_2	p_3	p_4	p_5	$f(.)$
u_8	1	0	0	0	0	1
u_9	1	0	0	0	0	1
u_{10}	1	0	0	0	0	1
u_{11}	1	1	1	0	0	3
u_{12}	1	0	1	0	1	3
u_{13}	1	0	0	0	1	2
u_{14}	1	0	0	0	1	2
u_{15}	1	1	1	1	0	4
u_{16}	1	1	1	1	0	4
u_{17}	1	0	0	0	0	1
u_{18}	1	1	1	1	0	4
u_{19}	1	0	1	0	1	3
u_{21}	1	0	0	0	0	1
u_{22}	1	0	0	0	0	1
u_{23}	0	1	1	0	0	2
u_{24}	0	1	1	0	0	2
u_{25}	1	0	0	0	0	1
u_{26}	0	1	1	0	0	2
u_{27}	1	0	0	0	0	1
u_{28}	0	1	1	0	0	2
u_{29}	1	0	0	0	0	1
u_{30}	0	1	1	0	0	2

4.2.3 Analysis of HPC Combination Reduction Based on Table 4.5

The reduction of Table 4.5 based on the proposed HPC combination algorithm is presented and discussed in this section to evaluate the performance of the porposed HPC algorithm. The algorithm can be improved by finding the means of new classification to generate solutions in enhancing the reduction cost. In solving the problem of parameter reduction, the decision partition order algorithm should use an intelligent method that has lower reduction but with better capability to effectively filter the parameters by using sufficient conditions to reduce the objects.

Table 4.5 Tabular representation of a soft-set

U/P	p_1	p_2	p_3	p_4	p_5	p_6	$f(.)$
u_1	1	1	1	0	0	1	4
u_2	1	1	1	0	1	0	4
u_3	1	1	1	0	1	0	4
u_4	0	1	1	0	0	1	3
u_5	1	1	1	1	1	0	5
u_6	1	1	1	1	0	1	5
u_7	0	0	0	0	1	1	2
u_8	0	0	1	1	1	0	3

4.3 Performance of Proposed Decision Partition Based on Probability

In order to evaluate and validate our decision partition order algorithm, the proposed Markov chain with respect to prioritized weight vector is implemented using Matlab 2010 simulation tool. We have implemented the proposed mathematical model of parameter reduction for decision partition order based on adjusted weight vector (in Chapter 3, Section 3.5), in order to mathematically prove the validity of our proposed algorithm and its ability in reduction. Figure 4.1 illustrates the achieved results that represent the probability of reduction using our proposed decision partition order algorithm in the form of probability tail exponatioal function and Probability Distribution Function (PDF). Figure 4.1a shows that our algorithm could achieve high fraction of total reduced parameters in probability tails of 0.9 when 16 parameters were given in a soft dataset. This indicates that our proposed decision partition order algorithm has efficient ability in reducing parameters of soft dataset up to 90% (14.4 of 16 parameters parameters were removed).

Figure 4.1b demonstrates the PDF of our proposed decision partition order algorithm with same number of parameters, (16) and random generated sub-set combinations. It is obvious that based on our proposed Markov chain model, our HPC algorithm could perform the best in terms of PDF in parameter reduction when the number of parameters and their random generated combinations are increased. Thus, a valid model was achieved by considering the weighted prioritized vector in the conducted decision making process of parameter reduction using proposed decision partition order algorithm.

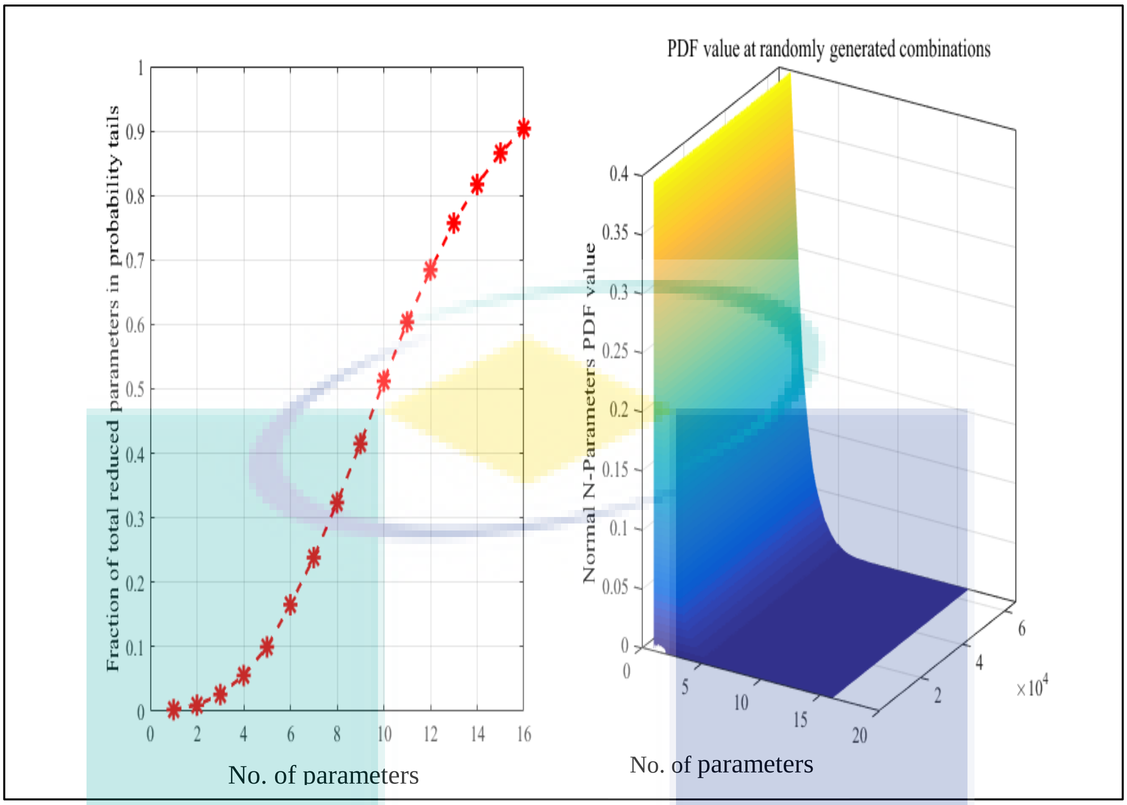

Figure 4.1a. Fraction tail probability of proposed PDF Decision partition order

Figure 4.1b Normal N-Parameter algorithm value

Figure 4.1 Probability validation of proposed Markov chain model of HPC algorithm

Figure 4.2 presents the modelled sequence of state transitions for parameter reduction using our proposed Markov chain model. We can observe the state transitions of 15 parameters (15 states) and their probability of reduction P_r within the selected 32 different combinations of sub sets. Utilizing the aforementioned methods in Section 3.5, which are adjusted weight vector and priority, the P_r of 15 parameters are simulated and presented, while in Figure 4.3, a clear interpretation of performance analysis of our proposed Markov chain model was given. The probability to be selected as the best sub-optimal decision in a percentage form is presented for each individual parameter. We can read from graphs in Figure 4.3 that parameters 10, 12 and 9 have obtained the highest probabilities respectively. In other words, parameters 10, 12 and 9 will be most highly recommended to remain as sub-optimal decisions in the final reduced soft set. It is worth noting that the reason behind this output is that, our proposed Markov chain model has considered the adjusted weight vectors and given priorities for each individual parameter in the soft set. The weight values were selected based in normalized weighted random numbers from 1 to 1000.

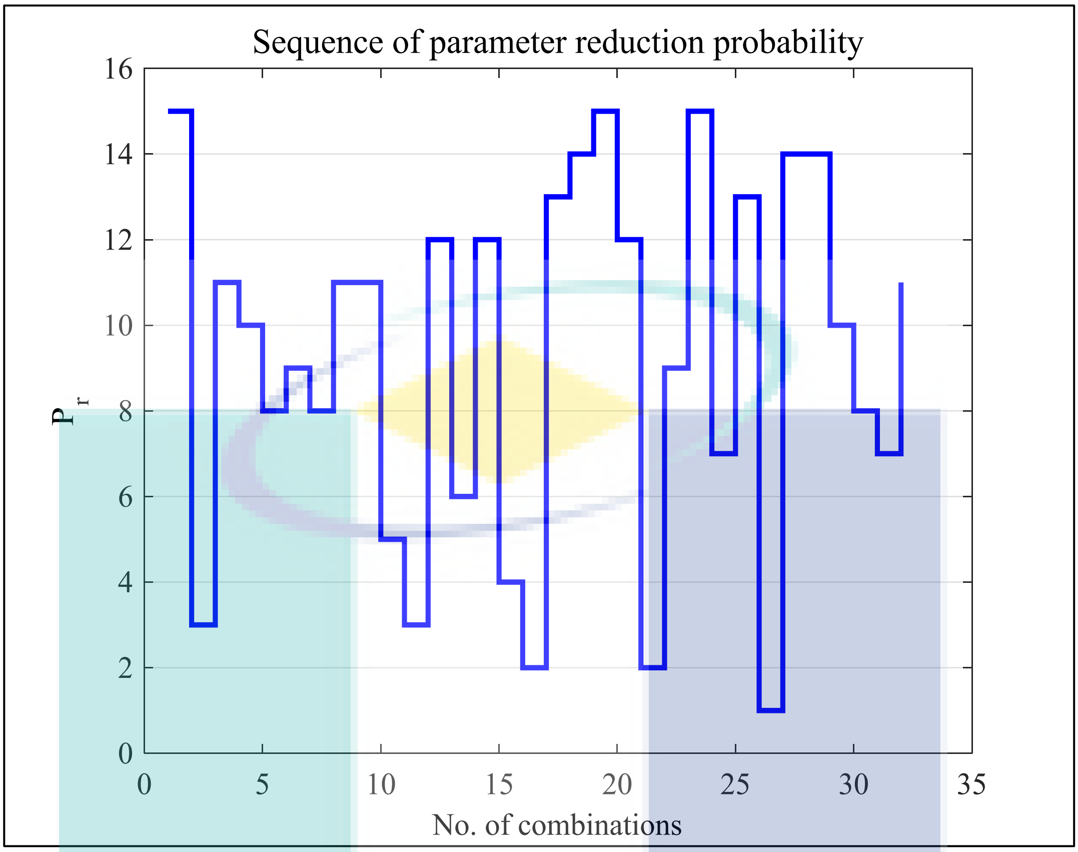

Figure 4.2 Sequence of state transitions for parameter reduction

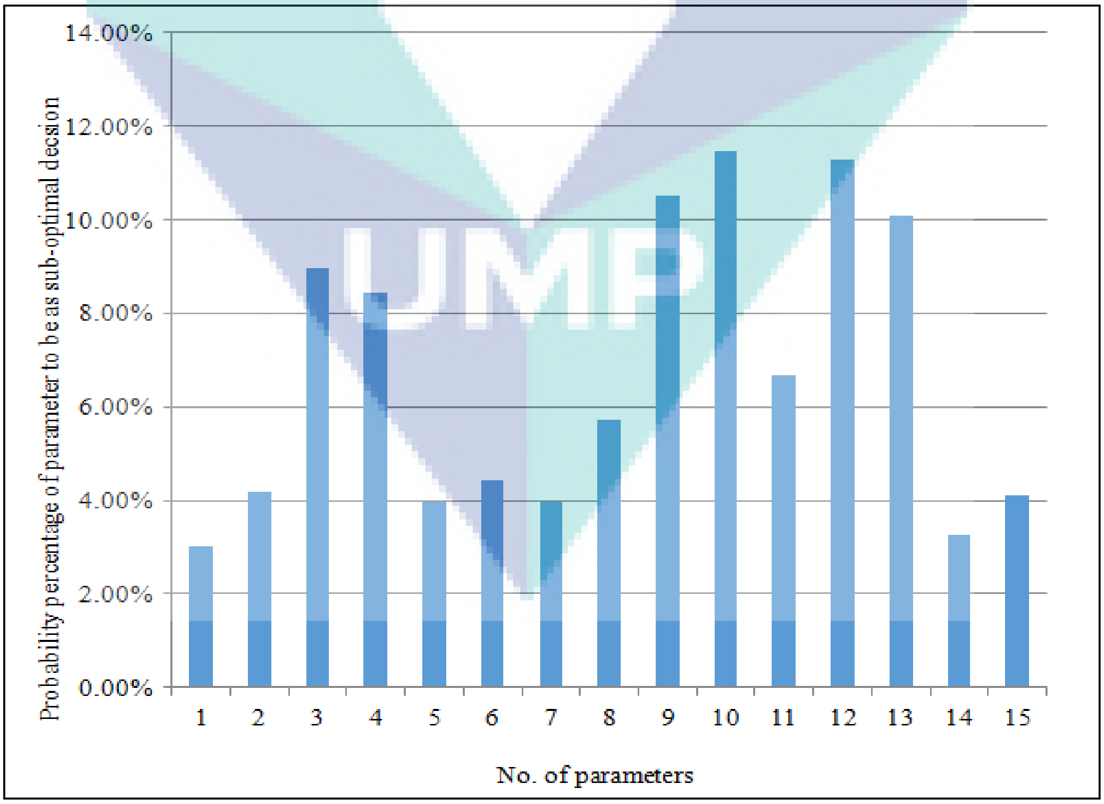

Figure 4.3 Probability percentage of the sub-optimal decision

4.4 Performance Evaluation of Proposed SSR-BPSO-BBO Algorithm for Predict Decision Partition Order

To evaluate our proposed SSR-BPSO-BBO algorithm a simulation was conducted based on Matlab 2010 tool. The binary version of PSO was implemented taking into account the recent improvements of particle's transfer functions V and S shaped that were proposed by Mirjalili et al. (2013). Moreover, for more accuracy, the BBO optimization algorithm was implemented on generated soft dataset as a way to reduce the average error of decision making of our proposed SSR algorithm. Thus, the lowest obtained error out of learning process of conducted Neural Network (NN) is considered on updating the velocity and position of each particle during searching process for the optimal decision. Hence, an optimized SSR algorithm was achieved in this study.

In order to prove the ability of BBO algorithm in classifying the given parameter in a softest data and showing its competency among the rest, it was implemented along with common classification algorithms to show their performance in classifying parameters of our dataset. In Figure 4.4, the classification rates based on Table 2.2 that were achieved by applying suggested BBO algorithm are compared with other common optimization algorithms, Particle Swarm Optimization (PSO), Genetic Algorithm (GA), Ant Colony Optimization (ACO), Evolutionary Strategy (ES) and Probability-Based Incremental Learning (PBEL)). It is important to mention that each algorithm was run 10 times and the average percentage is presented in Figure 4.4. It is clear enough that the suggested BBO based optimizer has achieved higher overall classification rate close to 100% compared with other optimization algorithms. The reason is that BBO has the best ability to avoid becoming trapped in local minima for the given dataset.

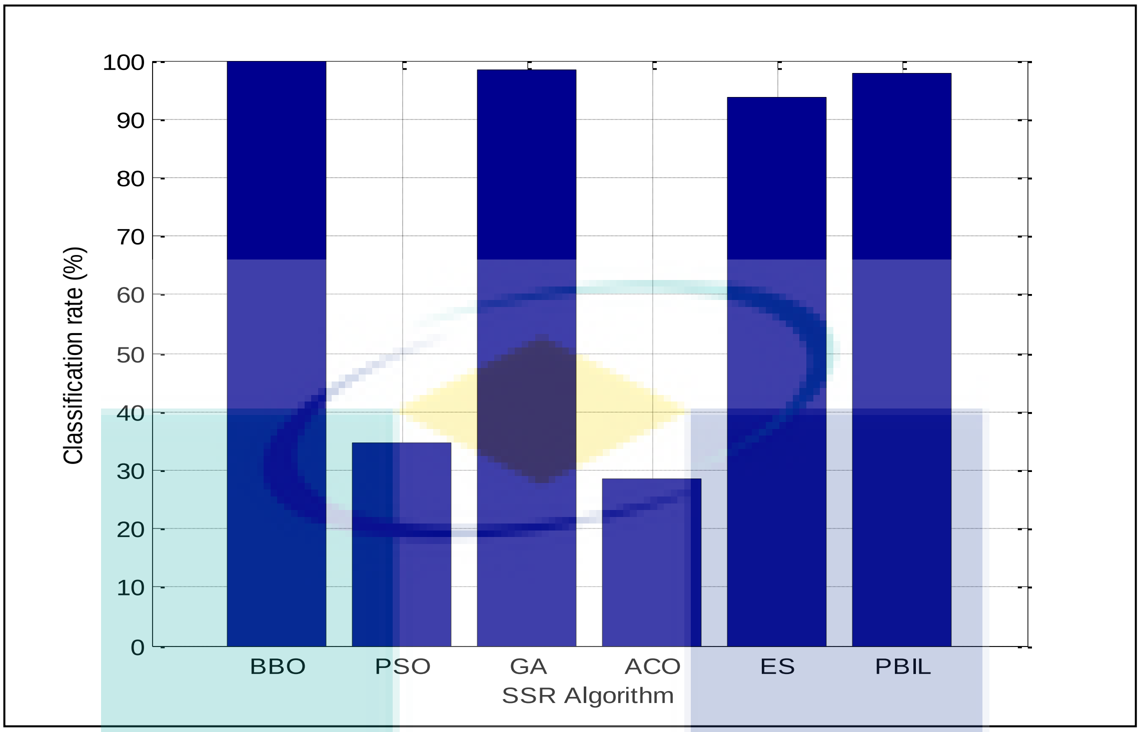

Figure 4.4 Classification rate of proposed SSR algorithm using BBO optimization algorithm compared with other optimization algorithms

SSE based on BBO could achieve minimum overall MSE compare with other algorithms as illustrated in Figure 4.5 BBO tends to have the fastest convergence behaviour on provided dataset. The experiment was run for 10 times with 300 generations in each. The average MSE was calculated after each run and presented in Figure 4.5 as overall average MSE. It is important to report here that the BBO algorithm has utilized the developed HSI of all habitats, which effects the growth in the initial random solutions for a particular problem scenario to obtain better convergence rate compared with other algorithms presented in Figure 4.5. Basically, BBO algorithm starts with a random set of habitats and every habitat has *n* different habitants that are correlated to the number of variables of an individual problem scenario. In addition, each habitat has its own immigration, emigration, and mutation rates. This simulates the typical geographically separated locations in nature.

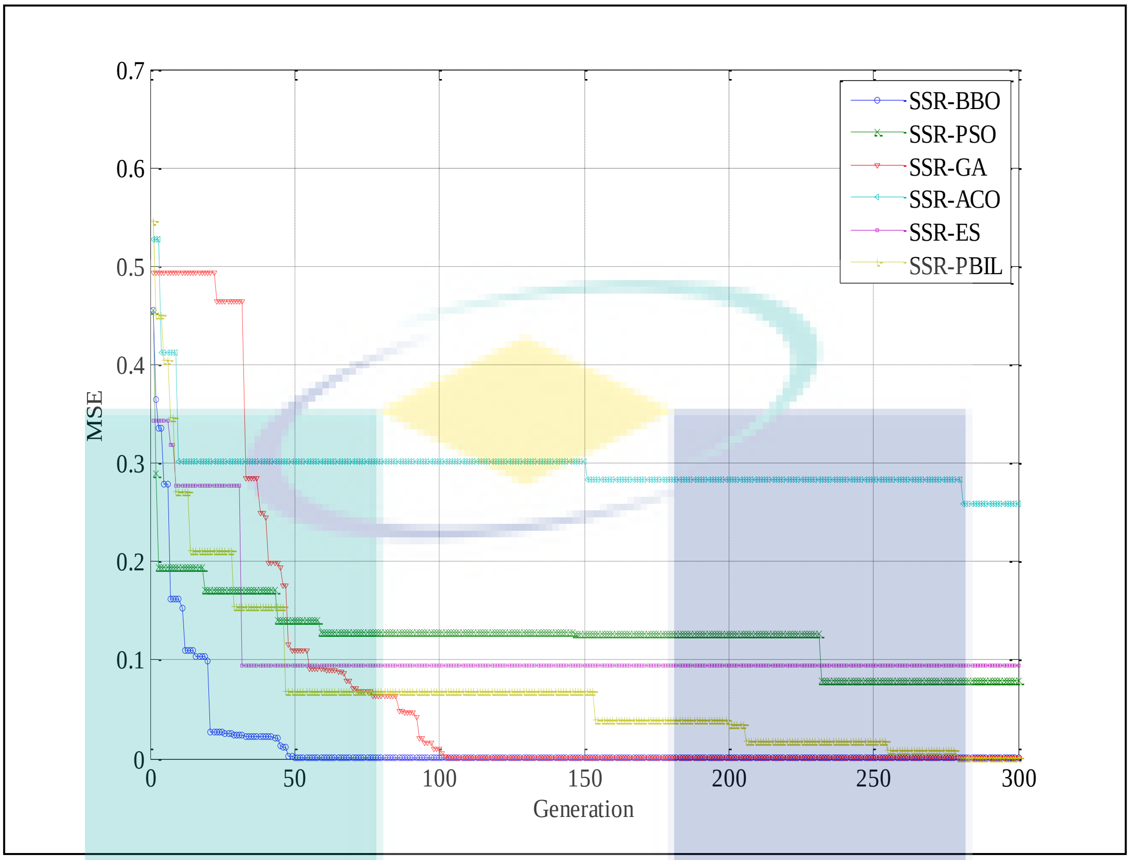

Figure 4.5 Convergence curves of MSE of SSR based BBO compared with other optimization algorithms of soft-dataset

The achieved results out of our proposed hybrid SSR-BPSO-BBO algorithm are presented in Figure 4.6. A soft dataset of eight (8) sub-sets and 200 parameters (v=200, refer to Table 8, Appendix A1) was generated randomly to test the ability of our proposed SSR-BPSO-BBO algorithm in selecting the optimal decision out of this big dataset during 600 general iterations of BPOS with assistance of 300 iterations of BBO optimization algorithm. We predefined a threshold called Average-of-the-Best (AoB) and we identified the parameter reduction criteria to be AoB ≥ 50. Hence, parameters that achieve AoB < 50 will be omitted from the list of optimal decisions.

It is clear that sub-set number 4 has obtained the best convergence curves in average compared with others. This is because sub-set 4 could maintain a steady behavior out of an average of 600 iterations. Sub-set 3 was the second ranked sub-set as it could achieve a lot of similar average best-so-far compared with sub-set 4, which represented 200 columns. Sub-set 7 has been removed since it does not satisfy our predefine constraint criteria. Sub-set 7 obtained an AoB less than 50 during the last few iterations, thus it has been excluded from the list of optimal decisions. We could achieve a highly significant optimized soft-set reduction algorithm that could address the weighty issue of data redundancy compared with existing algorithms. Below is the data of optimal decision (best option) obtained, which is sub-set number four (4).

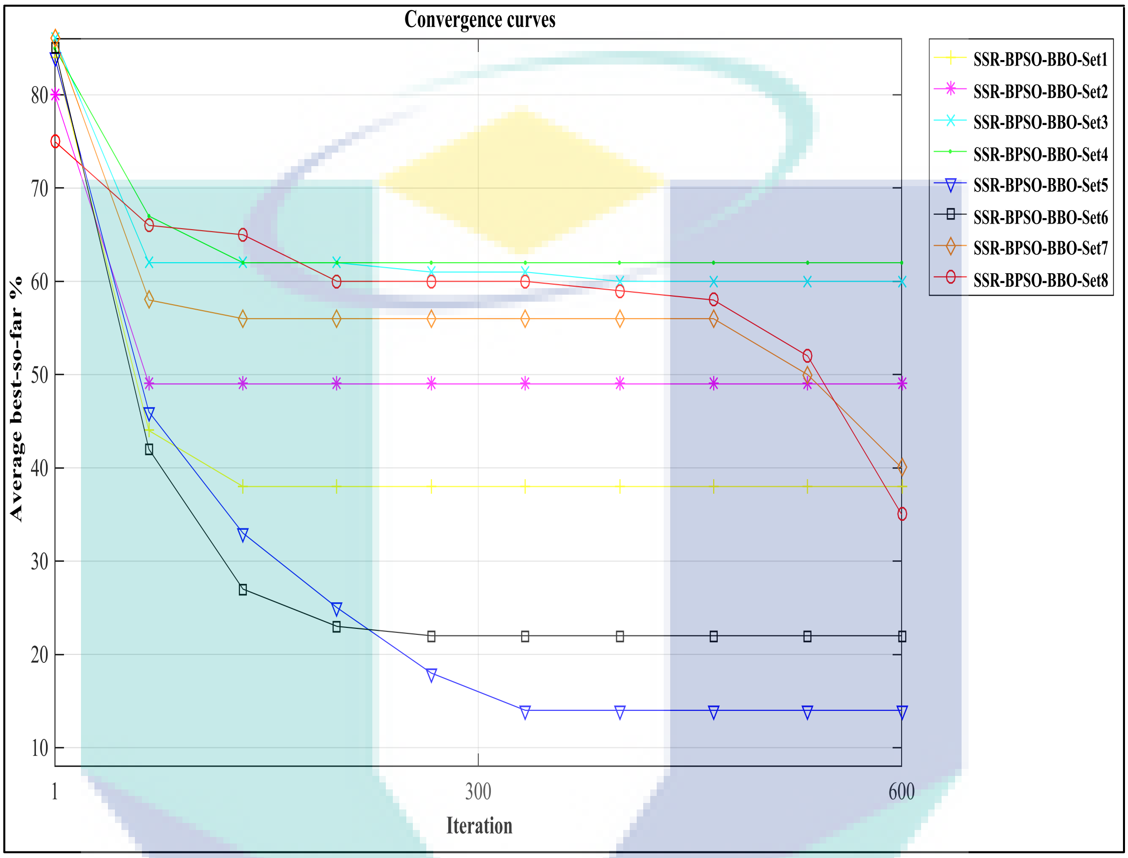

Figure 4.6 Convergence curves of Eight sub-sets of soft-dataset with v=200

4.5 Analysis Based on Decision Partition Order Algorithm

When the important data discovered the decision must be built based on the original properties, that should be free from uncertainty. The reason that, uncertainty generates inconsistency which represented by implies and non implies cases, which creates noise that is similar to email's spams situation. Our proposed decision partition order technique makes solid relations and ensures the consistency of original set while maintaining optimal and sub-optimal choices. The reduction rate increases using our proposed decision partition order technique based on priorities algorithm. The strength of the original property extractions include the existence of the relevant data that has relationship with the original.

4.5.1 Analysis Based on Decision Partition Order Algorithm in Table 4.5

The proposed decision partition order economically rationalising the original data into another version of data that can produce the same reduction ratio as obtained in Table 2.2 and Table 4.3. The proposed algorithm can improve the reduction by merging the data into smaller size, and it is flexible and it can be extended to other data properties such as priorities, which increases the discount rate and reduces memory usage and consequently reducing customer expense. For instance, the parameter reduction is achieved as follows:

The cluster of decision partition order for Table 4.5 generates partitions such as $\{\{u_1,u_2,u_3\},\{u_4,u_8\},\{u_5,u_6\},\{u_7\}\}$, which is categorized into $\{u_5,u_6\}$ as the maximum support (optimal decision partition order), $\{u_1,u_2,u_3\}$ as the second optimal decision partition, $\{u_4,u_8\}$ as the third optimal decision partition, and $\{u_7\}$ as the last optimal decision partition. The reduction obtained as a result of removing parameters p_3, p_5 and p_6 is 50% as shown in Table 4.6.

Table 4.6 Representation of the reduction soft set in Table 4.5

U/P	p_1	p_2	p_4	$f(.)$
u_1	1	1	0	2
u_2	1	1	0	2
u_3	1	1	0	2
u_4	0	1	0	1
u_5	1	1	1	3
u_6	1	1	1	3
u_7	0	0	0	0
u_8	0	0	1	1

4.6 Discussions of Proposed Characteristics within Computational Boundary

A low-cost choice of decision partitions would reduce the customer's expense. In this study, the customer and decision maker is provided with a set of algorithms for decision partitions with low choices rates and low computation while producing consistent decision quality, and the approach is extended to soft-set testing and evaluating the obtained solutions. The main benefit of reducing the length of choice is cost reduction, and that of decision quality is to allow significant candidate reduction while eliminating the arrival of false parameters at the candidate solutions. The soft weight is the pipeline or bridge between reduction process and selections criteria, which was assembled on similar object clusters. The basic concept is to group these clusters into disjoint groups as original characteristics. The following sections discuss the performance evaluation of the proposed algorithm and provide a comparison with other popularly used algorithms in the area of parameter reduction based soft-set.

4.6.1 Overall Decision Partition and Computational Cost

Optimal decision is an important business strategy that is determined with the best quality extension objective in reducing the overall cost under finite parameter. The minimum parameters were identified to completely fill the original decision partition in the reduction process, which is made possible with use of cluster and soft-set in arranging the subsets as well as determining the spacing between rows support. Small and large spacing of the support is determined based on cluster that is based on rough set equal to classless (similarity) rather than on the rough set lower and upper approximations involved in the parameter reduction. In next section, the analysis of HPC combination and the proposed complement are presented, followed by FPC combinations. The performance evaluation analysis is elaborated in this section, followed by the discussion on the alternative normal parameter reduction analysis.

Initially, during parameter reduction analysis, the characteristics of the original data should be assigned to computational cost, but the question is which characteristic should be assigned to a given boundary. The vagueness of some values inside the approximation range will be clarified during this analysis, which needs intelligent detections. To reduce the number of soft data, one can use set theory to analyze and simplify the huge data that helps to set unique original data characteristics. Several researchers have analyzed the original decision partition situation; however, some amount of vagueness cannot be waived. To solve this problem, one needs to study the characteristics of data and find their relations to determine the reduced data. Every soft reduction algorithm has to assign decision partition and their order properties. The verifications of the highest parameters reduction of independent decision partition order of the original set occurs at the maximum reduction, which is significantly close to the original characteristics boundary gap based on cluster and soft-set generated by the proposed Hybrid complement algorithm. Since the comparisons of any two decisions partition states based on equal classes is circular, this proves the decision partition order. In filtering false parameters, it is not necessary to check the complete original objects support cluster. It is sufficient to check only the Min supp and, consequently, the false parameters can be differentiated. Max supp is used to confirm the parameters that satisfy Min supp. Finally, R supp is used as a final check only to confirm those parameters that satisfy Min supp and Max supp clusters. The decision partition order algorithm is implemented in Java program as Net Beans IDE, 8.0.2 that is executed on Intel (R) Core (TM) 2 Duo CPU @.T6600@2.20 GHz running 32-bit operating system Windows 7 with 3.00 GB RAM.

Since the result of the reduction is determined based on combinations, the problem that needs to be solved is how to reduce the number of combinations, rather than reducing the full subset combinations of elements. To search for the decision solutions, the sub-combination has to be validated by pre-processing the parameters, but intelligent decision is required to identify where the solution is expected. However, the solution cannot be found if the computational boundary characteristic is not well-fitted, causing unnecessary loss of time and producing wrong results. The various algorithms used in this study performed the candidate boundary computational tests in finding the minimum boundary to produce accurate decision results.

4.6.2 Dimensionless Discussions of Decision Partition Order

Decision partition and its order is used to categorize soft data that qualify and perform reduction for generating new meaningful decisions. This process provides data object space that are separated into disjoint clusters such that every query has the smallest set of indexing block, which is an efficient query object accessing. The architecture describes the interaction between customers, decision makers (identified by a set of optimal choices levels and its relevant top ranking extracting) and the concept for each specified weight. The test reduction result are as shown in Table 2.2, Table 4.1, Table 4.3 and Table 4.5.

The parameterization reduction algorithm proposed by Kumar and Rengasamy (2013), was implemented for the purpose of benchmarking. It is worth noting that the proposed algorithm by Kumar deletes the clusters containing parameters that do not induce the original cluster, leaving the remaining parameter clusters to be checked looking for the optimal solution; the deleted parameters will be considered as a part of parameter reduction if their complement induced original decision partition. In our improved proposed algorithm, the clusters are checked for Min supp, Max supp and R supp. After the deletion process, if the induced Min supp resulting from the remaining parameters is not the same as the original Min supp, the deleted parameter is considered not part of the reduction, then kept; other wise ,Max check is required and if it satisfies the original Max partition, then R supp is checked to confirm whether or not R supp can be reduced.

Referring to Table 2.2, a comparison of the experimental results obtained based on previous algorithms and the proposed algorithms is shown in Figure 4.5. From the table, the dimensionless reductions are minimized using the following algorithms: Rose et al. (2010), Ma et al. (2011), Rose et al. (2012), Kumar and Rengasamy (2013), FPC combination, HPC combination, and complement based on decision partition order.

In Table 2.2, similar soft parameter reduction are performed on set $\{p_6\}$ as reduction that was deleted from $\{p_1, p_2, p_3, p_4, p_5, p_6\}$. The reduction dropped by up to 37% when using the algorithms proposed by Rose et al. (2012) and the HPC combination; meanwhile the reduction was 17% when using the algorithms suggested by Kumar and Rengasamy (2013), the FPC combination and the proposed complement algorithms. However, issues were encountered when applying Rose's (2012) algorithm due to object filtering ratio that increases the effect of reduction because the author overcame the false objects that have maximally or minimally supported parameters. Meanwhile, using the algorithms proposed by Rose et al. (2010) and Ma et al. (2011), on reducing Table 2.2, their co-efficiency becomes zero.

However, the proposed hybrid complement reduction method was updated in this study as a technique that works in two directions, first as a two-dimensional shape, then rotated for object reduction. Thus, the soft reduction rate for the updated version, as shown in Table 2.2, has increased up to approximately 37%, which is the same as that of Rose et al. (2012) and HPC proposed algorithm. The results shows that the object reduction issues is solved using the proposed complement method as the algorithm always produces reduction in both directions. As for objects, the reduction causes the complete cluster to be removed from sub-partitions. Also, from Table 2.2, the proposed complement algorithm, HPC combination and Rose's (2012) algorithm generate significant results as their average reduction rates are the highest, even on the parameters or objects as shown in Figure 4.7.

Also, referring to Table 2.2, the complement proposed algorithm based on decision partition order, HPC combination and (Rose et al., 2012) generate significant result as their average reduction rates are the highest, even on the parameters or objects side as shown in Figure 4.8.

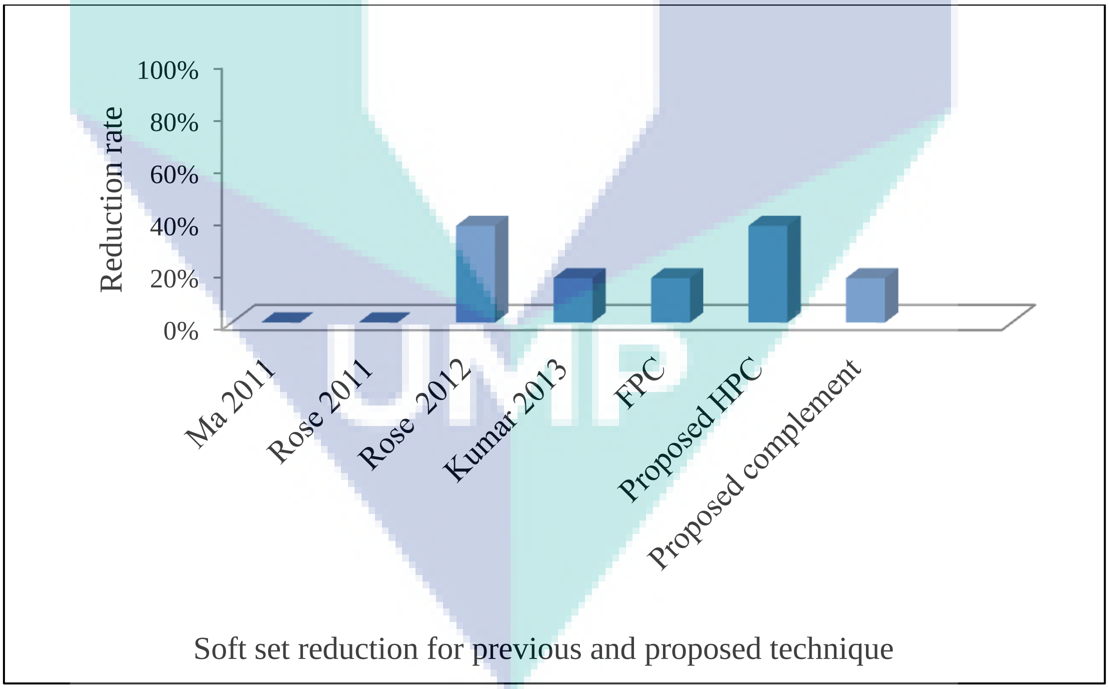

Figure 4.7 Represent reduction result of Set 2.2

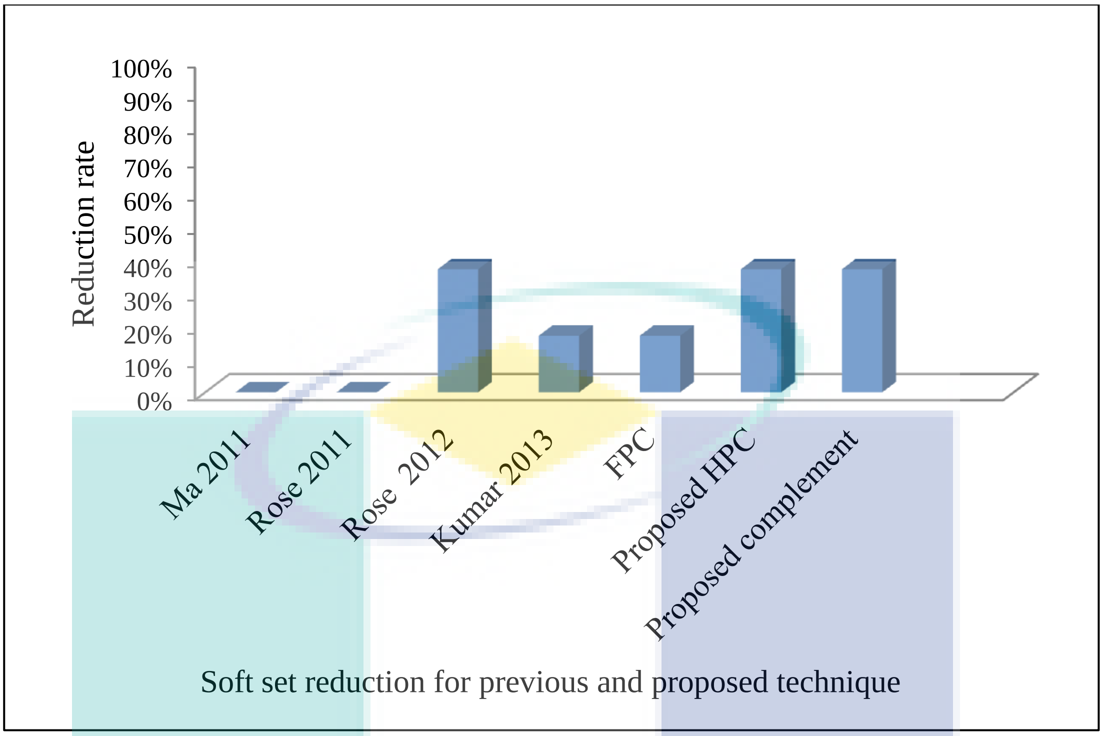

Figure 4.8 Improved reduction of Set 2.2

Referring to Table 4.3, the results obtained from checking the soft-set reduction based on previous algorithms and the proposed algorithms are shown in Figure 4.8, in which it was noted that the dimensionless reductions is minimized with algorithms suggested by Ma et al. (2011), Rose et al. (2012), Rose et al. (2012), and Kumar and Rengasamy (2013), and the FPC combination, HPC combination and the proposed complement method.

Based on Table 4.3, the measured soft parameter reduction obtained from all the algorithms, as shown in Figure 4.9, is moderate and the same reduction rate for all the algorithms after the parameters $\{p_6, p_7\}$ were deleted from the original set $\{p_1, p_2, p_3, p_4, p_5, p_6, p_7\}$. Except no result is obtained from the algorithm suggested Kumar and Rengasamy (2013). The reason for Kumar's algorithm not having any result is that the reduction of their implies is applied only when a single set is full or empty, and this has not reduced the density of implies that are represented in more than one column. Kumar and Rengasamy claimed that parameter reduction is determined based on a single parameter and they employed this property to calculate the false parameters, but if the parameters are individually deleted, then their implies condition cannot occur except when the parameter are maximally or minimally supported by parameters as in

Case 1. However, if the parameters in Case 1 have not been separately extracted and have implies, then the implies reduction would lead to an intermediate value that has a certain amount of uncertainly. Usually in the approach where every implies are not as in Case 1, the reduction results cannot be determined by depending on a single parameter.

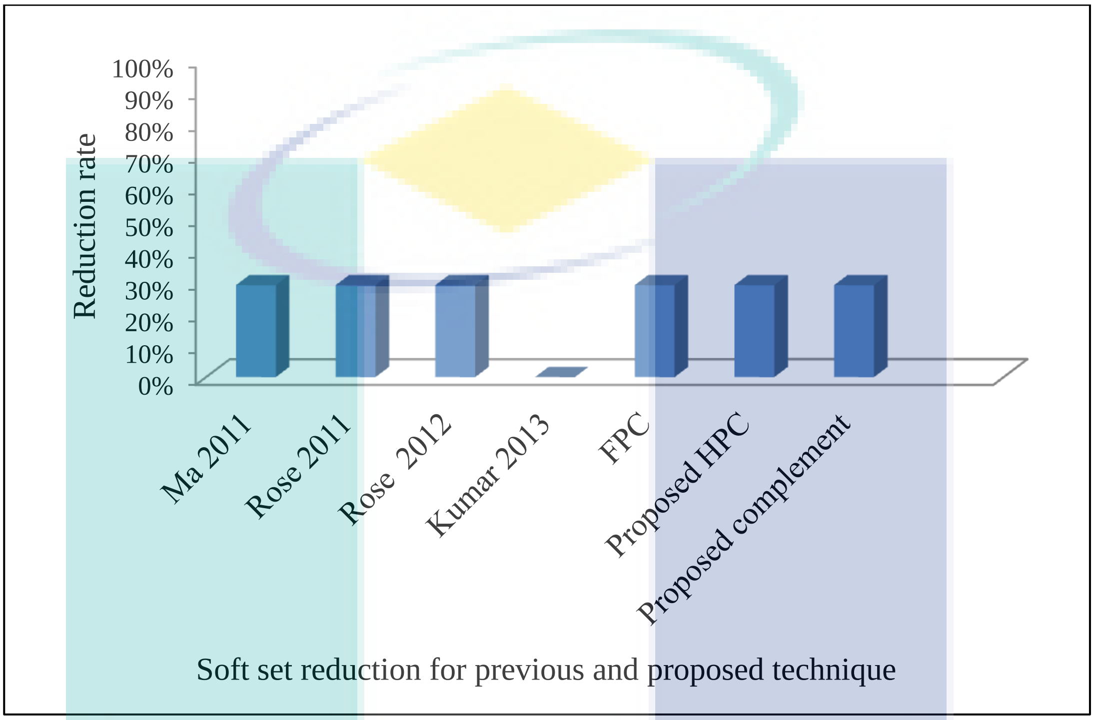

Figure 4.9 Reduction of Set 4.3

Based on Table 4.5 and using the algorithms mentioned, the soft parameter reduction rates obtained, after the decision partition order algorithm detected the parameters $\{p_3, p_5, p_6\}$ and deleted it from the original set $\{p_1, p_2, p_3, p_4, p_5, p_6\}$, is shown in Figure 4.10. The proposed complement algorithm achieved a reduction of approximately 50%, while no reduction is obtained from the algorithms suggested by Rose et al. (2010), Ma et al. (2011), Rose et al. (2012), Kumar and Rengasamy (2013), the proposed HPC algorithm and the FPC algorithm. Kumar and Rengasamy (2013) calculated false parameters reduction based on single parameter; however, no reduction of Table 4.5 will result if the parameters are individually deleted. Since implies condition do not occur in Table 4.5, even when the parameters are rotated, no reduction will be obtained from using the algorithms suggested by Rose et al. (2010), Ma et al. (2011), Rose et al. (2012), Kumar and Rengasamy (2013). FPC, and the proposed (HPC and complement) algorithms.

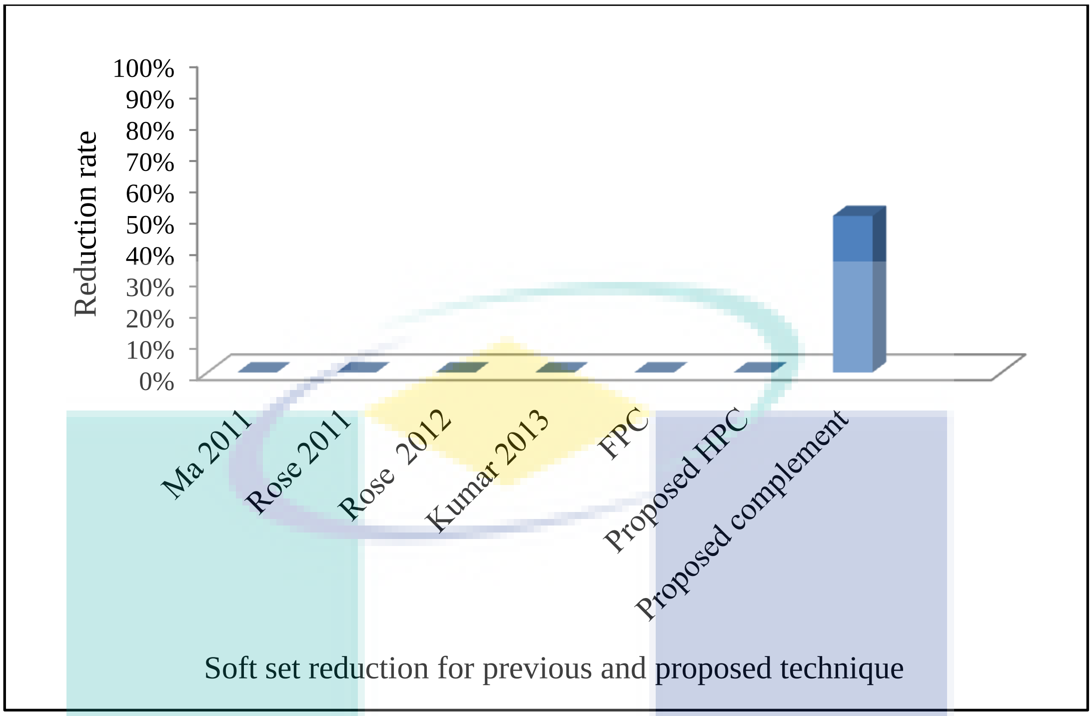

Figure 4.10 Reduction of Set 4.5

4.6.3 The Decision Partition Order Accuracy

With 200 parameters used to obtain decision partition order, 600 iterations were generated to predict the optimal decision partition (the object Set 4) and next sub-optimal decisions partitions (the object Set 3); the proposed SSR-BPSO-BBO performed the best by achieving 69% parameter reduction (removing 138 parameters out of 200 parameters) and 75% object or sub-set reduction (selecting optimal decisions 2 sub-sets out of 8).

4.7 Computational Result for Searching Strategy

Before searching where the solution is expected, the original characteristics need to be identified, which is done in prep-processing steps by decision partition order using indicernibility cluster in arranging the objects and maintaining the consistency of original characteristics. Based on Figure 4.11, in Part A, sub dimensional is decomposed to reduce implies (Herawan et al., 2009a) but not including single dimensional (Part B), while in Part B, Ma et al. (2011) proposed single dimensional reductions for implies if it is maximally or minimally supported by objects as Part B. Then, using the reduction is extended as multiple of N. Non-implies that are not reduced by the algorithms, Ma et al. (2011) and Kumar and Rengasamy (2013) produced decompositions that removed both Part B (implies) or Part C (non implies), if by removable their remaining parameters induced the original partitions. However, all non implies are not reduced by Kumar's algorithm if the non implies are placed in more than one column.

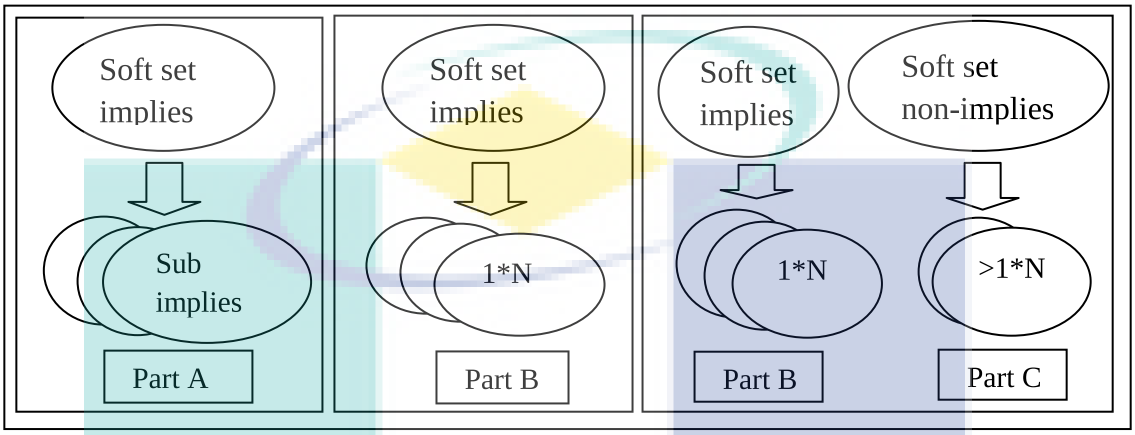

Figure 4.11 Represent reduction based on previous decompositions strategy

Referring to Table 4.5, the reduction obtained from checking the soft-set reduction based on previous decompositions algorithms and the proposed algorithms are shown in Figure 4.12. It is noted that the dimensionless reductions is not minimized with algorithms proposed by Ma et al. (2011), Rose et al. (2012), Rose et al. (2012), Kumar and Rengasamy (2013), and proposed Markov Model.

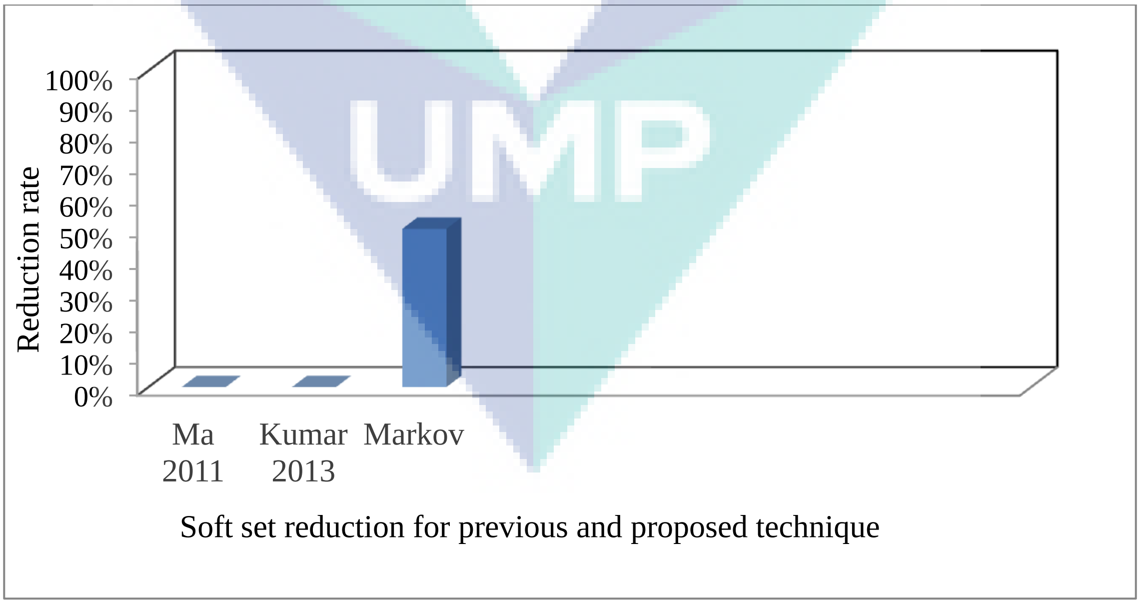

Figure 4.12 Represent space reduction based on Table 4.5

In the proposed algorithm based on Markov probability model in Table 4.5, the reduction is found after the sub reduction generated 20 spaces, while that of Ma et al. (2011) and Kumar and Rengasamy (2013) found no reduction even after checking 62 combinations. Taking into consideration the number of combinations that each algorithm needs to check as a computational cost in finding the best normal parameter reduction, the proposed algorithm could find the solution within 20 spaces; the computational cost obtained is (20/64) × 100% = 31.25%. On the other hand, the other algorithms failed to obtain any reduction even after checking all the possible 62 combinations, which means the computational cost is 100%. A comparison of the resulting computational cost is shown in Figure 4.13, and the use of the proposed Markov model is justified due to its improved reductions efficiency in terms of time.

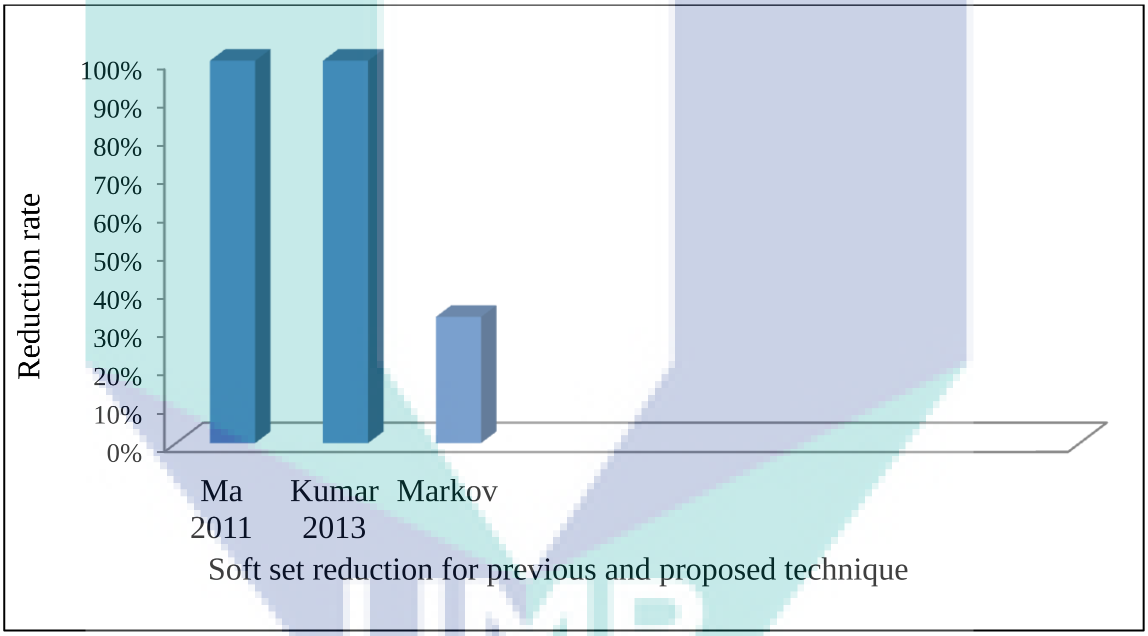

Figure 4.13 Computational cost based on Table 4.5

4.8 Summary

This chapter discusses the results obtained from the proposed SSR and HPC algorithms as well as that obtained from the algorithms proposed by Ma et al. (2011), Rose et al. (2012), Rose et al. (2012), Kumar and Rengasamy (2013), FPC combination and HPC combination. The proposed SSR and HPC algorithms are benchmarked against the other algorithms, it can be concluded that, based on the results obtained, all the objectives of the study as stated in Chapter 1 have been achieved. Furthermore, the capability of the algorithms to find the optimal decision is illustrated and discussed by applying the algorithms on given soft datasets. In particular, the proposed hybrid parameter complement reduction algorithm is used to reduce the computational cost of search domain that could help in managing uncertain data and maintaining the consistency during the process of parameter reduction. The first contribution of this study is to maintain consistency of optimal and sub-optimal choices by the use of the developed HPC algorithm. The improvement in the cost of parameter reduction benefited the customers and decision makers by producing smaller data size for transmission via communication lines for use in forecasting applications. The proposed HPC algorithm has also solved the infinite machine state and improved the decision characteristics classifications by setting the decision partition and its order. The process of fixing the characteristics of the original data has produced minimized choice cost with high data quality, which has also simplified the decision representations. The results obtained based on decision partition and its order has produced a new version of soft-set reduction algorithm that has uniquely addressed the issue of soft parameters reductions.

In addition, this chapter proposed two more contributions, namely, the development of mathematical model that could efficiently formulate the probability of parameter reduction based on adjusted weight vector and Markov Chain model, as well as the affirmation of data classifications via the proposed SSR-BPSO-BBO algorithm that utilize the soft-set data classification to obtain the optimal decision. In first component this study contributes to maintain consistency of optimal and sub-optimal choices by the use of developed SSR-BPSO-BBO algorithm.

Object or sub-set reduction has also addressed in this chapter, which was achieved by rotating two-dimensional tables using the proposed hybrid complement SSR-BPSO-BBO algorithm to generate candidate objects solutions; the proposed algorithm has improved sub-set reduction by approximately 10% compared with that of the state-of-the-art algorithm. Furthermore, the proposed SSR-BPSO-BBO algorithm could efficiently address the issue of choosing an optimal decision out of a large number of parameter combinations, which was achieved by considering the S and V shaped transfer functions based on binary version of PSO that was optimized via BBO to achieve low MSE of the obtained decision. Thus, a significant contribution was achieved by obtaining a finite answer as the optimal decision.

CHAPTER 5

CONCLUSION

5.1 Introduction

Numerous studies have been conducted in favor of data reduction to ensure high quality and integrity of data before it can be processed in supporting decision making. In the last decades, several algorithms were developed in the field of data reduction that are aimed at extracting valuable information from a large collection of data, in which case the reduction process is very important since the human brain has limitations in performing decision making. The human brain is only able to make decisions based on specific amount of information or choices, but cannot extend the decision beyond the provided information. Soft theory shows that the coefficient of Boolean classifications can compress vital digital data closer into a smaller space.

There are several parameter reduction techniques that deal with uncertainty in soft-set theory to achieve reduction based on information characteristics in reducing implies and some non-implies condition, but the techniques failed to provide consistency in both implies and non-implies. This study has reviewed previous studies on the effect of soft-set reduction techniques on parameter reduction that were conducted by previous researchers such as Maji et al. (2002), Chen et al. (2005), Kong et al. (2008), Rose et al. (2010), Mamat et al. (2011), Rose et al. (2012) and Kumar and Rengasamy (2013). However, the concept used in their studies does not consider some implies and non implies as reduction; thus, if the implies or non-implies are placed in more than one column, the algorithms can no longer capture it.

This current review is useful in enhancing characteristic of decision partition classifications that are not stable, however, these cases do not maintain the accuracy of the classifications. Another issue is that the reduction efficiency is not enhanced if the domains of sub sets are not feasible due to large parameters. The large parameters induced infinite combinations, and for this reason previous studies have decomposed the parameters into small sub sets by removing the complexity of feasible domain but this does not generate optimal decision.

The proposed algorithm in this research is significant to the body of knowledge as it could efficiently maintain the optimal and sub-optimal choices during the process of soft set reduction. Moreover, the proposed SSR-BPSO-BBO algorithm will efficiently address the issue of obtaining an optimal decision with high classification rate and less error.

In this research three main contributions are proposed, which are the hybrid technique that is used for reducing the computational cost of search domain for managing uncertain data and maintaining the consistency during parameter reduction process, the development of mathematical model that could efficiently formulate the probability of parameter reduction based on adjusted weight vector and Markov Chain model, and maintaining soft-set data classifications via the proposed SSR-BPSO-BBO algorithm that utilise the soft-set data classification to obtain the optimal decision.

The first component of this study contributes to maintain consistency of optimal and sub-optimal choices by the use of the developed HPC algorithm. The enhancement in the achieved cost of parameter reduction has benefited the customers and decision makers by producing smaller data size that can be easily sent through transmission lines. The performance of the proposed HPC algorithm has solved the infinite machine state and improved the decision characteristics classifications by setting the decision partition and its order. The process of fixing the original characteristics has minimized choice cost by producing high data quality, which has also simplified the decision representations. The results obtained, as shown in Chapter 4, which is based on decision partition and its order has produced a new version of soft-set reduction algorithm that has uniquely addressed the issue of soft parameters reductions.

In addition, the object or sub-set reduction has also been addressed in this study, which is achieved via two stages, first by rotating two-dimensional tables using hybrid complement algorithm to generate candidate objects, and then the candidate object solutions measured the significance of the candidate solutions by using Jacquard similarity against the decision partition and its order to remove complete sub-sets. Consequently, the steps enhanced the proposed hybrid complement algorithms, which increased the reduction by up to 37% instead of 17% achieved by other algorithms. Moreover, the decision partition order algorithm has improved sub-set reduction by approximately 10% compared to that obtained using Rose's algorithm (Rose et al., 2012). As a result, the sub-set reduction process has significantly improved as a part of soft-set reduction process using the proposed decision partition order.

The proposed probability model of parameter reduction based on adjusted weight vector and Markov Chain model has validated the efficiency of the proposed parameter reduction algorithm in achieving high probability of reduction with respect to the increase in the number of parameters and their possible random combinations. The proposed SSR-BPSO-BBO algorithm could efficiently address the issue of choosing an optimal decision out of a huge number of parameter combinations. This was achieved by considering the S and V shaped transfer functions based on binary version of PSO, which was optimized via BBO to achieve low MSE of obtained decision. Thus, a significant contribution was achieved by obtaining a finite answer as the optimal decision.

The next section presents the summarized computational cost of the proposed algorithms. Finally, recommendations for future commotional reductions is discussed.

5.2 Summarized Computational Cost

The proposed probability algorithm will first remove the dispensable parameters. It can be concluded that implies decompositions are multiples of N and will close the infinite machine state at (total 1's or total 0's) by 1*N, 2*N, 3*N,...(M-1)*N, where M and N the numbers of objects and parameters, respectively. Also, the decomposition gap of non implies is closed at (total 1's+total 0's) if the implies are decomposed into 1*N, 2*N, 3*N,...(M-1)*N. The decomposition process is followed with the construction of candidates by reconstructing the result obtained from the decomposition process. During this process, any decomposed implies will be reduced if all of its objects supports are the same, and any decomposed non implies will be reduced if their complement objects decision partition induced the original objects decision partition order.

In terms of the decomposition used for enhancing the search domain, the proposed probability algorithms has reduced the soft-set by 32% compared to 0% obtained from existing algorithms. This improvement in parameter reduction is the result of using the probability model instead of removing one-dimensional parameters for removing non implies as suggested by Kumar (2013); removing one-dimensional will overcome the parameters that are maximally or minimally induced by objects. The solutions is first executed to reduce the search domain, which will enhance decompositions time by reducing the search domain before decomposition process. Finally, the efficiency of decompositions enhances the infinite machine state and improves the checking of further reductions.

The proposed algorithm implemented in this study will also enhance the decomposition process by reducing the decompositions time as continuous checking is avoided. Thus, optimization takes place by eliminating specific parameters from the overall soft-set combination. The proposed algorithm has speeded up the time required to obtain the candidate solutions by reducing the parameter reduction process. Thus, the proposed SSR-BPSO-BBO algorithm is more efficient and performs better in obtaining more accurate optimum decision by overcoming the difficulties and limitations of existing algorithms.

5.3 Summary of Original Characteristics Findings

The findings obtained from the analysis of the soft data is presented in this study, in which less choices is employed in dimensionality reduction in finding the minimum number of significant parameter, objects or sub sets. The technique used is able to identify the false frequent data that has negative effect on the reduction process, and in searching the domain, the decomposition process is able to improve computational cost and remove complexity type.

The need to maintain the original classifications while reducing the parameters was one of the problems faced when using several previous soft-set reduction techniques, while high decision cost decreases the quality of information and produces inconsistency. The existing algorithms provide the solutions for soft parameter reduction but their solutions are not assigned full classifications as their reduction are partially solved. The findings from using the proposed algorithm is that it enhances the reduction quality, controls the relationship among the information, and improves the original soft-set the decision quality and choices cost.

This research has found a method to determine the classifications of the original set by setting decision partition cluster and decision partition cluster order. The objects decisions supports decompositions improved in terms of searching efforts in checking candidate's solution, in which the enhancement of objects support decompositions have been made by Min supp, Max supp and R supp. As mentioned earlier, existing algorithms suffer from the possible occurrence of false frequent during reduction. Thus, to find reliable soft-set reduction, the proposed Hybrid parameterization reduction algorithm has implemented decision partition cluster and decision partition cluster order criteria checking in order to generate reduction.

The proposed decision partition order method can be used to check reduction inconsistency, similar to other previous algorithms proposed by Ma (2011), Rose et al. (2012), Kumar and Rengasamy (2013), FPC combination and HPC combination. Using Kumar's algorithm, FPC combination and HPC combination, the reduction achieved is 37%, while a reduction of up to 37% is obtained when using the proposed hybrid complement algorithm. The proposed hybrid complement method attained higher reduction because it reduces the parameters as well as objects, and the reduction is generated in each rotation. Ma's and Rose's algorithm cannot perform reductions in the compressions, and Kumar's algorithm cannot perform reduction when the non implies or the implies occur in more than one column; however, the proposed hybrid complement method is able to generate reduction and the accuracy of decision partition is predicted using SSR-BPSO-BBO.

5.4 Summary of Minimum Candidate Solutions

The decomposition process proposed in this study is able to solve the problem of infinite machine state by generating finite solutions, and exact solutions are produced with minimum searching effort. The complexity of candidate boundary is solved by removing or selecting sub sets, and the decompositions of the original characteristics induced Min supp, Max supp and R supp that eliminated some candidates from candidate sets. Several algorithms, such as that proposed by Ma, Kumar and the proposed method, could generate reduction, but the benefit of the decomposition stage is to decrease the number of checking for the solutions. The total number of combinations and transfer methods of the proposed method minimises the search domain to produce results at minimum complexity, however, the methods suggested by Ma and Kumar do not always generate reductions. Using Ma's algorithm, no reduction will occur if non implies is present as the reduction is overcome by search domain reduction, while Kumar's algorithm could not detect any reductions in candidate solutions if implies occur.

5.5 Recommendations and Future Work

It is recommended that a compact and an alternative technique soft-set reduction algorithm be developed to produce exact solutions in terms of choice cost reduction while maintaining the characteristics of the original data. The proposed combinations could not handle dispensability status when it's infinite parameters are placed in more than one column. Therefore, it will open new horizons to make use of existing techniques at the same time creative, innovative and genius solutions should be made. In future studies, it would be interesting to investigate the efficiency of the proposed algorithm by implementing it into other research fields such as networks sensors, flood and disaster forecasting:

The algorithm to be developed should generate solutions in enhancing candidate solutions reduction complexity for the following reasons. The algorithm should use intelligent methods and at the same time has lower complexity in solving the problem of parameter reduction in a more compact method. It is also recommended that future researchers should attempt to find an innovative method for determining solutions for soft-set or other types of data, in which the reduction should be rely on more powerful algorithms to ensure that quality decision can be achieved in infinite parameters as well as the accuracy of the original consistency.

REFERENCES

Abedjan, Z., & Naumann, F. (2011). Advancing the discovery of unique column combinations. In *Proceedings of the 20th ACM International Conference on Information and Knowledge Management (CIKM),* 1565-1570.

Acıkgoz, A., & Tas, N. (2016). Binary soft set theory. *European Journal of Pure and Applied Mathematics*, 9(4), 452-463.

Adeli, H., & Hung, S.L. (1994). An adaptive conjugate gradient learning algorithm for efficient training of neural networks. *Applied Mathematics and Computation*, 62(1), 81-102.

Akerkar, R., & Sajja, P. (2010). *Knowledge-Based Systems*. MA, USA: Jones & Bartlett.

Alkhazaleh, S. (2015). The multi interval valued fuzzy soft set with application in decision Making. *Applied Mathematics*, 6(8), 1250-1262.

Asemi, A., Safari, A., & Zavareh, A.A. (2011). The role of Management Information System and Decision Support System for manager's decision making process. *International Journal of Business and Management*, 6(7), 164-173.

Awang, M.K., Makhtar, M., Rahman M.N.A., & Deris, M. M. (2016). A new soft set based pruning algorithm for ensemble method. *Journal of Theoretical and Applied Information Technology*, 88(3), 384-391.

Ayyub, B.M., & Klir, G.J. (2006). *Uncertainty Modeling and Analysis in Engineering and the Sciences*. Boca Raton, Florida: Chapman & Hall, CRC Press.

Babitha, K.V., & Sunil, J.J. (2010). Soft set relations and functions. *Computers & Mathematics with Application*, 60(7), 1840-1849.

Bakshi, T., Sinharay, A., & Som, T. (2016). A Novel soft set theoretic prisoners' dilemma game model. In *Proceedings of the 3rd IEEE International Conference on Recent Advances in Information Technology (RAIT),* 160-163.

Baluja, S. (1994). *Population-Based Incremental Learning: A Method for Integrating Genetic Search Based Function Optimization and Competitive Learning* (Report No. CMU-CS-94-163). Pittsburgh, PA: Carnegie-Mellon University.

Bell, D.E. (1982). Regret in decision making under uncertainty. *Operations Research*, 30(5), 961-981.

Bernstein, D.S. (2005). *Matrix Mathematics: Theory, Facts, and Formulas with Application to Linear Systems Theory*. Princeton, New Jersey: Princeton University Press.

Blum, C., & Socha K. (2005). Training feedforward neural networks with ant colony optimization: An application to pattern classification. In N. Nedjah, L. Mourelle, M. Vellasco, A. Abraham, & M. Koppen (Eds.), *Proceedings of the 5th IEEE International Conference on Hybrid Intelligent Systems (HIS)*, 233-238.

Boussaid, I., Lepagnot, J., & Siarry, P. (2013). A survey on optimization metaheuristics, *Information Sciences, 2*37, 82-117.

Branke, J. (1995). Evolutionary algorithms for neural network design and training. In J.T. Alander (Eds.), *Proceedings of the 1st Nordic Workshop on Genetic Algorithms and its Applications (NWAGA)*, 154-163.

Cagman, N., & Enginoglu S. (2010). Soft matrix theory and its decision making. *Computers & Mathematics with Applications*, 59(10), 3308- 3314.

Chang, M.Y., Hung, Y.C., Yen, D.C., & Tseng, P.T. (2009). The research on the critical success factors of knowledge management and classification framework project in the executive Yuan of Taiwan Government. *Expert Systems with Applications*, 36(3), 5376-5386.

Chen, D., Tsang, E.C.C., Yeung, D.S., & Wang, X. (2005). The parameterization reduction of soft sets and its applications. *Computers & Mathematics with Applications*, 49(5), 757-763.

Chen, Y-C., Shang, R-A., & Kao, C-Y. (2009). The effects of information overload on consumers' subjective state towards buying decision in the internet shopping environment. *Electronic Commerce Research and Applications*, 8(1), 48-58.

Chen, W., & Zou, Y. (2016). A hybrid method for decision making with dependence & feedback under incomplete information. In *Proceedings of* the *3rd SHS Web Conference on Information Technology and Career Education (ICITCE)*, 134-140.

Citkin, V., Aygunoglu, A., & Aygun, H. (2016). A new approach in handling soft decision making problems. *Jornal of Nonlinear Science & Application*, 9, 231-239.

Csaji, B.C. (2001). *Approximation with Artificial Neural Networks.* (Master Thesis). Faculty of Sciences, Etvs Lornd University, Hungary.

Dalkir, K., & Liebowitz, J. (2011*). Knowledge Management in Theory and Practice*. Burlington, Massachusetts: Elsevier Butterworth-Heinemann.

Damghani, K.K., Taghavifard, M., & Moghaddam, R.T. (2009). *Decision Making Under Uncertain and Risky Situations*. Schaumburg, Illinois: Symposium Monograph Society of Actuaries.

Danjuma, S., Herawan, T., Ismail, M.A., Chiroma, H., Abubakar, A.I, & Zeki, A.M. (2017). A Review on soft set-based parameter reduction and decision making. *IEEE Access*, 5, 4671-4689.

Dasgupta, D., & Michalewicz, Z. (1997). *Evolutionary Algorithms in Engineering Applications*. Berlin: Springer-Verlag.

Deli, I., & Cagman, N. (2016). Application of soft sets in decision making based on game theory. *Journals of Fuzzy Mathematics and Informatics*, 11(3), 425-438.

Demetrovics, J. (1980). On the equivalence of candidate keys with sperner systems. *Acta Cybernetica*, 4(3), 247-252.

Deng L., & Wiebe, J. (2015). Joint Prediction for entity/event-level sentiment analysis using probabilistic soft logic models. In *Proceedings of the 15th Conference on Empirical Methods in Natural Language Processing (EMNLP)*,179-189.

Dymek, G., & Walendziak, A. (2015). Soft set theory applied to general algebras. *Demonstration Mathematica*, 48(1), 13-20.

Edwards, J. (2011). Process view of knowledge management: It ain't what you do, it's the way that you do it. *The Electronic Journal of Knowledge Management*, 9(4), 297-306.

Feng, F., Cho J., Pedrycz, W., Fujita, H., & Herawan, T. (2016). Soft set based association rule mining. *Knowledge-Based Systems*, 111, 268-282.

Feng, F.L.Y., & Cagman, N. (2012). Generalized uni-int decision making schemes based on choice value soft sets. *European Journal of Operational Research*, 220(1), 162-170.

Fernandez-Diaz, A., Baier, C., Benac-Earle, C. & Fredlund, L.A. (2012). Static partial order reduction for robabilistic concurrent systems. In *Proceedings of the 9th IEEE International Conference on Quantitative Evaluation of Systems (QEST)*, 104-113.

Fertier, A., Montarnal, A., Barthe-Delanoe, A.M., Truptil, S., & Benaben, F. (2016). Adoption of big data in crisis management toward a better support in decision making. In A. Tapia, P. Antunes, V.A. Banuls, K. Moore, & J. Porto (Eds.), *Proceedings of the 13th International Conference on Information Systems for Crisis Response and Management (ISCRAM)*, 66-72.

Fine, T.L. (1999). *Feedforward Neural Network Methodology*. New York: Springer Science & Business Media.

Fraenkel, A.A., Bar-Hillel, Y., & Levy, A. (1973). *Foundations of Set Theory*. Burlington, MA: Elsevier.

Fulmer, C.A. (2011). *Developing Information Storage and Retrieval Systems on the Internet a Knowledge Management Approach.* (Doctoral Dissertation). Naval Postgraduate School, Monterey, California.

Gardner, M.W., & Dorling, S.R. (1998). Artificial neural networks (the multilayer perceptron): A review of applications in the atmospheric sciences and atmos. *Atmospheric Environment,* 32(14-15), 2627-2636.

Gobithaasan, R.U., Hassan, N., Miura K.T., & Suzuki, H. (2016). Product design with soft set theory. In Salleh et. al (Eds.), *Proceedings of the 23rd Malaysian National Symposium on Mathematical Science (SKSM*23*)*, 1750, 1-7.

Gottschalk, P. (2007). *Knowledge Management Systems in Law Enforcement: Technologies and Techniques*. Hershey, Pennsylvania: Idea Group Inc.

Gudise, V.G., & Venayagamoorthy, G.K. (2003). Comparison of particle swarm optimization and back propagation as training algorithms for neural networks. In *Proceedings of the 1st IEEE Conference on Swarm Intelligence Symposium (SIS)*, 110-117.

Guo, Z.X., Wong, W.K., & Li, M. (2012). Sparsely connected neural network based time series forecasting. *Information Sciences*, 193, 54-71.

Hakim, R.B.F., Sari, E.N., & Herawan, T. (2014). On if-then multi soft sets-based decision making. In Linawati et. al (Eds.), *Proceedings of the 2nd EurAsia Conference on Information and Communication Technology (ICT-EurAsia)*, 306-315.

Herawan, T. (2014). *Recent Advances on Soft Computing and Data Mining*. Switzerland: Springer International Publishing.

Herawan, T., Rose, A.N.M., & Deris, M.M. (2009a). Soft set theoretic approach for dimensionality reduction. *Database Theory and Application*, 171-178.

Herawan, T., & Deris, M.M. (2009b). A direct proof of every rough set is a soft set. In *Proceedings of the 3rd Asia International Conference on Modelling & Simulation (AMS)*, 119-124.

Herawan, T., & Deris, M.M. (2009c). A soft Set Approach for Association Rules Mining. *Knowledge Based System*, 24(1), 186-195.

Hinton, G.E., and Sejnowski, T.J. (1999). *Unsupervised Learning: Foundations of Neural Computation.* Cambridge: The MIT Press.

Hoppner, F. (1999). *Fuzzy Custer Analysis: Methods for Classification, Data Analysis and Image Recognition*. New York: John Wiley & Sons.

Hoque, A.S.M., Halder, P.K., Parvez, M.S, & Szecsi, T. (2013). Integrated manufacturing features and design for manufacture guidelines for reducing product cost under CAD/CAM environment. *Computers & Industrial Engineering*, 66(4), 988-1003.

Hornik, K., Stinchcombe, M., & White, H. (1989). Multilayer feedforward networks are universal approximators. *Neural Networks*, 2(5), 359-366.

Hush, D.R., & Bill G.H. (1993). Progress in supervised neural networks. *IEEE Signal Processing Magazine*, 10(1), 8-39.

Ibrahim, A.M, & Yusuf, A.O. (2012). Development of soft set theory. *America International Journal of Contemporary Research,* 2(9), 205-210.

Jaccard, P. (1902). Distribution comparee de la flore alpine dans quelques regions des Alpes occidentales et orientales. *Bulletin de la Sociale*, 31, 81-92.

Ji, M., Han, J., & Danilevsky, M. (2011). Ranking-based classification of heterogeneous information networks. In *Proceedings of the 17th ACM International Conference of Special Interest Group on Knowledge Discovery and Data Mining (SIGKDD)*, 1298-1306.

Jiang, J.J., Klein, G., Wu, S.P., & Liang, T.P. (2009). The relation of requirements uncertainty and stakeholder perception gaps to project management performance. *The Journal of Systems and Software*, 82(5), 801-808.

Junco, J.G.D., Zaballa, R.D.R., & Perea, J.G.A. (2010). Evidence based administration for decision making in the framework of knowledge strategic management. *The Learning Organization*, 17(4), 343-363.

Kantardzic, M. (2011). *Data Mining: Concepts, Models, Methods, and Algorithms.* Vol XVII, 2nd Ed. New Jersey: Wiley-IEEE Press.

Kennedy, J., & Eberhart, R.C. (1997). A discrete binary version of the particle swarm algorithm. In *Proceedings of the 10th IEEE International Conference on Systems, Man and Cybernetics (SMC)*, 4104-4108.

King, W.R. (2009). Knowledge Managementand Organizational Learning. In *Knowledge Management and Organizational Learning.* New York: Springer Science & Business Media.

Kıvan, M., Mıhcak, Akyol, E., Basar, T., & Langbort, C. (2016). Scalar quadratic Gaussian soft watermarking games. In *Proceedings of the 7th International Conference on Decision and Game Theory for Security (GameSec)*, 215-234.

Kong, Z., Gao, L., Wang, L., & Li, S. (2008). The normal parameter reduction of soft sets and its algorithm. *Computers & Mathematics with Applications*, 56(12), 3029-3037.

Kong, Z., Jia, W., Zhang, G., & Wang, L. (2015). Normal parameter reduction in soft set based on particle swarm optimization algorithm. *Journal of Applied Mathematical Modelling*, 39(16), 4808-4820.

Karaaslan, F. (2016). Soft Classes and Soft Rough Classes with Applications in Decision Making. *Mathematical Problems in Engineering*, 2, 1-11.

Kumar, U.S., & Inbarani, H.H. (2015). A Novel neighborhood rough set based classification approach for medical diagnosis. *Procedia Computer Science*, 47, 351-359.

Kumar, D.A., & Rengasamy, R. (2013). Parameterization reduction using soft set theory for better decision making. In *Proceedings of the 9th IEEE International Conference on Pattern Recognition, Informatics and Mobile Engineering (PRIME)*, 365-367.

Laudon, K.C, & Laudon, J.P (2004). *Management Information Systems: Managing the Digital Firm*. New Jersey: Prentice Hall.

Lee, J., & Wang, Y. (2016). *On the Semantic Relationship between Probabilistic Soft Logic and Markov Logic*. Cornell University, New York: arXiv preprint.

Ma, H., Simon, D., Fei, M., & Xie, Z. (2013). Variations of biogeography based optimization and Markov analysis. *Information Sciences*, 220, 492-506.

Ma, X., Sulaiman, N., Qin, H., Herawan, T., & Zain, J.M. (2011). A new efficient normal parameter reduction algorithm of soft sets. *Computers & Mathematics with Applications*, 62(2), 588-598.

Maier, R. (2007). *Knowledge Management Systems: Information and Communication Technologies for Knowledge Management*. 3rd Ed. Berlin: Springer-Verlag.

Maji, P., Roy, A.R., & Biswas, R. (2002). An application of soft sets in a decision making problem. *Computers & Mathematics with Applications*, 44(8), 1077-1083.

Mamat, R., Herawan, T., & Deris, M.M. (2011). Super attribute representative for decision attribute selection. In J.M. Zain, W.M.W. Mohd, & E. El-Qawasmeh (Eds.), *Proceedings of the 2nd International Conference on Software Engineering and Computer Systems(ICSECS), Part* 1, 137-147.

McCulloch, W.S., & Pitts, W. (1943). A logical calculus of the ideas immanent in nervous activity. *The Bulletin of Mathematical Biophysics*, 5(4), 115-133.

Melin, P.P., Sanchez, D., & Castillo, O. (2012). Genetic optimization of modular neural networks with fuzzy response integration for human recognition. *Information Science*, 197, 1-19.

Mendes, R., Cortez, P., Rocha, M., & Neves, J. (2002). Particle swarms for feedforward neural network training. In *Proceedings of the 12th IEEE International Joint Conference on Neural Networks (IJCNN)*, 1895-1899.

Merminod, V., & Rowe, F. (2012). How does PLM technology support knowledge transfer and translation in new product development? Transparency and boundary spanners in an international context. *Information and Organization*, 22(4), 295-322.

Milligan, G.W., & Cooper, M.C. (1985). An examination of procedures for determining the number of clusters in a data set. *Psychometrika*, 50(2), 159-179.

Min, H., & Eom, S.B. (1994). An integrated decision support system for global logistics. *International Journal of Physical Distribution & Logistics Management*, 24(1), 29-39.

Mirjalili, S., & Andrew, L. (2013). S-shaped versus V-shaped transfer functions for binary particle swarm optimization. *Swarm and Evolutionary Computation*, 9, 1-14.

Mirjalili, S., & Hashim, S.Z.M. (2010). A new hybrid PSOGSA algorithm for function optimization. In *Proceedings of the 1st IEEE International Conference on Computer and Information Application (ICCIA)*, 374-377.

Mirjalili, S., Lewis, Andrew, & Ali, S.S. (2014). Autonomous particles groups for particle swarm optimization. *Arabian Journal for Science and Engineering*, 39(6), 4683-4697.

Mirjalili, S., & Hashim, S.Z.M. (2012). BMOA: Binary Magnetic Optimization Algorithm. *International Journal of Machine Learning and Computing*, 2(3), 204-208.

Mirjalili, S., Mirjalili, S.M., & Andrew, L. (2014). Let a biogeography-based optimizer train your multi-layer perceptron. *Information Sciences*, 269, 188-209.

Mitchell, T.M., Keller, R.M., & Kedar-Cabelli, S.T. (1986). Explanation based generalization: A unifying view. *Machine Learning*, 1(1), 47-80.

Mohamad, M., & Selamat, A. (2016). A New hybrid rough set and soft set parameter reduction method for spam e-mail classification task. In *Proceedings of the 14th Pacific Rim Knowledge Acquisition Workshop (PKAW)*, 18-30.

Molodtsov, D. (1999). Soft set theory—first results. *Computers & Mathematics with Applications*, 37(4-5), 19-31.

Moschovakis, Y.N. (2009). *Descriptive Set Theory*. 2nd Ed. Providence, Rhodes Island: American Mathematical Society.

Muller, H., & Freytag, J-C. 2005. *Problems, Methods, and Challenges in Comprehensive Data Cleaning*. (Report No. iroB-1B-164)). Humboldt-Universitt zu Berlin: Institute fur Informatik.

Ngai, E.T., Xiu, L., & Chau, D.C. (2009). Application of data mining techniques in customer relationship management: A literature review and classification. *Expert Systems with Applications*, 36(2), 2592-2602.

Norris, J, R. (1998). *Markov Chains*. Cambridge: Cambridge University Press.

Oja, E. (2002). Unsupervised learning in neural computation. *Theoretical Computer Science*, 287(1), 187-207.

Olson, D.L., & Delen, D. (2008). *Advanced Data Mining Techniques*. New York: Springer Science & Business Media.

Onyeozili, I.A., & Gwary, T.M. (2014). A Study of the fundamentals of soft set theory. *International Journal of Scientific & Technology Research*, 3(4), 132-143.

Osei-Bryson, K-M., Mansingh, G., & Rao, L. (2014). *Knowledge Management for Development: Domains, Strategies and Technologies for Developing Countries*. Berlin: Springer-Verlag.

Padmanabhan, S., Malkemus, T., Jhingran, A., & Agarwal, R. (2001). Block oriented processing of relational database operations in modern computer architectures. In *Proceedings of the 17th IEEE International Conference on Data Engineering (ICDE)*, 567-574.

Pant, J., Juyal A., & Bahuguna, S. (2015). Soft set: A soft computing for dimensionality reduction. *International Journal of Innovative Science, Engineering & Technology*, 2(4), 728-735.

Parmee, I.C. (2001). *Evolutionary and Adaptive Computing in Engineering Design*. New York: Springer Science& Business Media.

Pawlak, Z. (1991). *Rough Sets: Theoretical Aspects of Reasoning About Data*. Dordrecht, Netherlands: Kluwer Academic Publishers.

Pawlak, Z. (1998). Rough set theory and its applications to data analysis. *Cybernetics & Systems*, 29(7), 661-688.

Pawlak, Z., Grzymala-Busse, J., Slowinski, R., & Ziarko, W. (1995). Rough sets. *Communications of the ACM*, 38(11), 88-95.

Pawlak, Z., & Skowron, A. (2007). Rudiments of rough sets. *Information Sciences*, 177(1), 3-27.

Polkowski, L. (2013). *Rough Sets: Mathematical Foundations*. Berlin: Springer-Verlag.

Poo, D., Kiong, D., & Ashok, M.S. (2008). *Object-Oriented Programming and Java*. 2nd Ed. London: Springer.

Puccinelli, N.M., Goodstein, R.C., Grewal, D., Price, R., Raghubir, P., & Stewart, D. (2009). Customer experience management in retailing: understanding the buying process. *Journal of Retailing*, 85(1), 15-30.

Ravindran, S., & Iyer, G.S. (2014). Organizational and knowledge management related antecedents of knowledge use: The moderating effect of ambiguity tolerance. *Information Technology and Management*, 15(4), 271-290.

Reed, R.D., & Marks, R.J. (1998). *Neural Smithing: Supervised Learning in Feedforward Artificial Neural Networks*. Cambridge: MIT Press.

Rose, A.N.M., Awang, M.I., Hassan, H., Zakaria, A. H., Herawan, T., & Deris, M.M. (2012). Hybrid reduction in soft set decision making. In *Proceeding of the 7th International Conference on Intelligent Computing (ICIC)*, 108-115.

Rose, A.N.M., Herawan, T., & Deris, M.M. (2010). A framework of decision making based on maximal supported sets. In *Proceeding of the 7th International Symposium on Neural Networks (ISNN)*, 473-482.

Rosenblatt, F. (1957). *The Perceptron: A perceiving and Recognizing Automation* (Report No. 85-460-1). Ithaca, New York: Project PARA, Cornell Aeronautical Laboratory.

Saaty, T.L. (1994). *Fundamentals of Decision Making and Priority Theory with the Analytic Hierarchy Process*. Vol VI, 2nd Ed. Pittsburgh University, Pennsylvania: RWS Publications.

Saraf, S. (2013). Survey or review on soft set theory and development. *The Standard International Journals Transactions on Computer Science Engineering & Its Applications*, 1(3), 59-66.

Seiffert, U. (2001). Multiple layer perceptron training using genetic algorithms. In *Proceedings of the 9th European Symposium on Artificial Neural Networks (ESANN),* 159-164.

Selvachandran, G., & Salleh, A.A. (2016). Fuzzy parameterization intuititionistic fuzzy soft expert set theory and its applications in decisions making. *International Journal of Soft Computing*, 11(2), 52-63.

Simon, D. (2008). Biogeography-based optimization. *IEEE Transactions on Evolutionary Computation*, 12(6), 702-713.

Solanki, S. (2016). Related study of soft set and its application a review. *Innovative Systems Design and Engineering*, 7(4), 12-22.

Sutoyo, E., Mungad, M., Hamid, S., & Herawan, T. (2016). An efficient soft set based approach for conflict analysis. *PLoS One*, 11(2), 1-31.

Tripathy, B.K., & Arun, K.R. (2015). A new approach to soft sets, soft multisets and their Properties. *International Journal of Reasoning-Based Intelligent Systems*, 7(3-4), 244-253.

Ulucay, V., Oztekin, O., Sahin, M., Olgun, N., & Kargin, A. (2016). Soft representation of soft groups. *New Trends in Mathematical Sciences*, 2, 23-29.

Vaduva, I. (2012). On Solving some types of multiple attribute decision making problems. *Romanian Journal of Economic Forecasting*, 15(1), 41-61.

Vaidya, J., & Clifton, C. (2002). Privacy preserving association rule mining in vertically partitioned data. In *Proceedings of the 9th ACM International Conference of Special Interest Group on Knowledge Discovery and Data Mining (SIGKDD),* 639-644.

Walczak, B., & Massart, D. (1999). Rough sets theory. *Chemometrics and Intelligent Laboratory Systems*, 47(1), 1-16.

Wang, G., Guo, L., Wang, H., Duan, H., Liu, L., & Li, J. (2014). Incorporating mutation scheme into krill herd algorithm for global numerical optimization. *Neural Computing & Applications*, 24(3-4), 853-871.

Wang, G-G., Gandomi, A.H., & Alavi, A.H. (2013). A chaotic particle swarm krill herd algorithm for global numerical optimization. *Kybernetes*, 42(6), 962-978.

Wang, G-G., Gandomi, A.H., & Alavi, A.H. (2013). An effective krill herd algorithm with migration operator in biogeography-based optimization. *Applied Mathematical Modelling*, 38(9), 2454-2462.

Wang, G-G., Gandomi A.H., Alavi, A.H., & Hao, G-S. (2013). Hybrid krill herd algorithm with differential evolution for global numerical optimization. *Neural Computing & Application*, 25(2), 297-308.

Wang, G-G., Gandomi, A.H., & Alavi, A.H. (2014). Stud krill herd algorithm. *Neurocomputing*, 128, 363-370,

Werbos, P.J. (1974). *Beyond Regression: New tools for prediction and analysis in the behavioural sciences*. (Doctoral Dissertation). Applied Mathematics, Harvard University, MA.

Werbos, P.J. (1992). Neurocontrol and Supervised Learning: An Overview and Evaluation. In *Handbook of Intelligent Control* (65-89). New York: Van Nostrand Reinhold.

Wienholt, W. (1993). Minimizing the system error in feedforward neural networks with evolution strategy. In *Proceedings of the 3rd International Conference on Artificial Neural Network (ICANN)*, 490-493.

Xu, W., Ma, J., Wang, S., & Hao, G. (2010). Vague soft sets and their properties. *Computers & Mathematics with Applications*, 59(2), 787-794.

Xu, W., Xiao, Z., Dang, X., Yang, D., & Yang, X. (2014). Financial ratio selection for business failure prediction using soft set theory. *Knowledge-Based Systems*, 63, 59-67.

Xu, W., & Xiao, Z., (2016). Soft set theory oriented forecast combination method for business failure prediction. *Journal of Information Processing System*, 12(1), 109-128.

Yang, D., Xiao Z., Xua, W., Wang, X., & Pan, Y. (2016). A Novel soft set approach for feature selection. *International Journal of Database Theory and Application*, 9(5), 77-90.

Yao, X. (1993). Evolutionary artificial neural networks. *International Journal of Neural System*, 4(3), 203-222.

Yu, H., Huang, X., Hu, X., & Wan, C. (2009). Knowledge management in e-commerce: A data mining perspective. In *Proceedings of the 2nd IEEE International Conference on Management of e-Commerce and e-Government (ICMECG)*, 152-155.

Zhang, N. (2009). An online gradient method with momentum for two-layer feedforward neural networks. *Applied Mathematics & Computation*, 212(2), 488-498.

Zhan, J., Liu, Q., & Herawan, T. (2017). Anovel soft rough set: Soft rough hemirings and corresponding multicriteria group decision making. *Applied Soft Computing*, 54, 393-402.

Zhang, Q., & Wang, X. (2016). A New parameter reduction method based on soft set theory. *International Journal of Hybrid Information Technology*, 9(5), 99-108.

Zhao, Y., Luo, F., Wong, S.M., & Yao, Y. (2007). A general definition of an attribute reduct. In *Proceedings of the 2nd International Conference on Rought Set and Technology (RSKT)*, 101-108.

Zhu, P., & Wen, Q. (2010). Probabilistic soft sets. In *Proceedings of the 6th IEEE International Conference on Granular Computing (GrC)*, 635-638.

Zolhavarieh, S., Aghabozorgi, S., & Teh, Y.W. (2014). A review of subsequence time series clustering. *The Scientific World Journal*, 1, 1-19.

APPENDIX A

A SOFT SET DATA TABLE 200

Table 2.8 Soft set representations

S/V	v_1	v_2	v_3	v_4	v_5	v_6	v_7	v_8	v_9	v_{10}
s_1	0	0	0	0	0	1	0	0	0	0
s_2	0	0	0	0	0	0	1	0	1	0
s_3	1	0	0	0	1	0	0	1	0	0
s_4	1	0	0	0	0	1	0	0	0	0
s_5	0	0	0	0	0	0	0	1	1	0
s_6	0	0	0	0	0	0	0	0	1	0
s_7	0	0	0	0	0	0	0	0	0	0
s_8	0	0	0	0	0	0	0	0	0	0
$S(v_j)$	2	0	0	0	1	2	1	2	3	0

Table 2.8 Continued.

S/V	v_{11}	v_{12}	v_{13}	v_{14}	v_{15}	v_{16}	v_{17}	v_{18}	v_{19}	v_{20}
s_1	0	0	0	0	0	1	0	1	0	0
s_2	0	0	1	1	1	0	0	0	0	0
s_3	0	0	0	1	0	0	0	0	1	0
s_4	0	0	1	0	1	1	0	1	1	1
s_5	0	0	0	0	0	0	0	0	0	0
s_6	0	0	0	0	0	0	0	0	0	0
s_7	0	0	0	0	0	0	0	0	0	0
s_8	0	0	0	0	0	0	0	0	0	0
$S(v_j)$	0	0	2	2	2	2	0	2	2	1

Table 2.8 Continued.

S/V	v_{21}	v_{22}	v_{23}	v_{24}	v_{25}	v_{26}	v_{27}	v_{28}	v_{29}	v_{30}
s_1	1	1	0	0	0	0	1	0	0	1
s_2	0	1	0	0	0	0	0	0	0	0
s_3	0	0	0	0	1	0	0	0	0	0
s_4	0	0	0	0	1	0	1	0	0	0
s_5	0	0	0	0	0	0	0	0	0	0
s_6	0	0	0	0	0	0	0	0	0	0
s_7	0	0	0	0	0	0	0	0	0	0
s_8	0	0	0	0	0	0	0	0	0	0
$S(v_j)$	1	2	0	0	2	0	2	0	0	1

Table 2.8 Continued.

S/V	v_{31}	v_{32}	v_{33}	v_{34}	v_{35}	v_{36}	v_{37}	v_{38}	v_{39}	v_{40}
s_1	0	0	0	0	0	0	0	0	0	0
s_2	0	0	0	0	0	0	0	0	1	0
s_3	0	0	1	1	0	0	0	1	0	1
s_4	0	0	0	0	0	0	0	0	0	1
s_5	0	0	0	0	0	0	0	0	0	0
s_6	0	1	0	0	0	0	0	0	0	0
s_7	0	0	0	0	0	0	0	0	0	0
s_8	0	0	0	0	0	0	0	0	0	0
$S(v_j)$	0	1	1	0	0	0	0	1	1	2

Table 2.8 Continued.

S/V	v_{41}	v_{42}	v_{43}	v_{44}	v_{45}	v_{46}	v_{47}	v_{48}	v_{49}	v_{50}
s_1	0	1	0	0	0	0	0	0	0	0
s_2	1	1	0	1	0	1	0	1	1	0
s_3	1	0	0	0	0	0	1	0	0	1
s_4	0	1	0	1	0	1	0	0	0	1
s_5	0	0	0	0	0	0	0	0	0	0
s_6	0	0	0	0	0	0	0	0	0	0
s_7	0	0	0	0	0	0	0	0	0	0
s_8	0	0	0	0	0	0	0	0	0	0
$S(v_j)$	2	3	0	2	0	2	1	1	1	2

Table 2.8 Continued.

S/V	v_{51}	v_{52}	v_{53}	v_{54}	v_{55}	v_{56}	v_{57}	v_{58}	v_{59}	v_{60}
s_1	0	0	1	0	0	1	1	0	0	0
s_2	1	0	1	0	0	0	0	1	0	0
s_3	0	0	0	0	0	0	0	0	0	0
s_4	0	0	0	1	0	0	0	0	1	1
s_5	0	1	0	0	0	1	0	0	1	0
s_6	0	0	0	0	0	0	0	0	0	0
s_7	0	0	0	0	0	0	0	0	0	0
s_8	0	0	0	0	0	0	0	0	0	0
$S(v_j)$	1	1	2	1	0	2	1	1	2	1

Table 2.8 Continued.

S/V	v_{61}	v_{62}	v_{63}	v_{64}	v_{65}	v_{66}	v_{67}	v_{68}	v_{69}	v_{70}
s_1	1	0	1	1	0	0	1	0	0	0
s_2	0	1	0	0	0	0	1	0	0	0
s_3	0	0	1	1	1	1	1	1	0	0
s_4	0	0	0	0	1	0	0	1	1	0
s_5	0	0	0	0	0	0	0	0	0	0
s_6	0	0	0	0	0	0	0	0	0	0
s_7	0	0	0	0	0	0	0	0	0	0
s_8	0	0	0	0	0	0	0	0	0	0
$S(v_j)$	1	1	2	2	2	1	3	2	1	0

Table 2.8 Continued.

S/V	v_{71}	v_{72}	v_{73}	v_{74}	v_{75}	v_{76}	v_{77}	v_{78}	v_{79}	v_{80}
s_1	0	0	1	0	1	0	0	0	0	0
s_2	1	0	0	0	1	1	0	0	0	1
s_3	1	1	0	0	1	0	0	0	0	0
s_4	0	0	0	1	0	0	0	1	0	1
s_5	0	0	0	0	0	0	0	0	0	0
s_6	0	0	0	0	0	0	0	0	0	0
s_7	0	0	0	0	0	1	0	0	0	0
s_8	0	0	0	0	0	0	0	0	0	0
$S(v_j)$	2	1	1	1	3	2	0	1	0	2

Table 2.8 Continued.

S/V	v_{81}	v_{82}	v_{83}	v_{84}	v_{85}	v_{86}	v_{87}	v_{88}	v_{89}	v_{90}
s_1	0	1	0	1	0	0	0	0	0	0
s_2	1	0	1	0	0	0	1	0	1	0
s_3	0	0	0	0	0	1	1	0	0	1
s_4	0	0	0	0	1	0	0	1	1	0
s_5	0	0	0	0	0	0	0	0	0	0
s_6	0	0	0	0	0	0	0	0	0	0
s_7	0	0	0	0	0	0	0	0	0	0
s_8	0	0	0	0	0	0	0	0	0	0
$S(v_j)$	1	1	1	1	1	1	2	1	2	1

Table 2.8 Continued.

S/V	v_{91}	v_{92}	v_{93}	v_{94}	v_{95}	v_{96}	v_{97}	v_{98}	v_{99}	v_{100}
s_1	0	0	0	0	1	0	0	0	0	0
s_2	0	0	0	0	1	0	0	0	0	0
s_3	0	0	0	1	1	1	0	1	0	0
s_4	0	0	0	1	0	0	1	0	0	1
s_5	0	0	0	0	0	0	1	0	0	0
s_6	0	0	0	0	0	0	0	0	0	0
s_7	0	0	0	0	0	0	0	0	0	0
s_8	0	0	0	0	0	0	0	0	0	0
$S(v_j)$	0	0	0	2	3	1	2	1	0	1

Table 2.8 Continued.

S/V	v_{101}	v_{102}	v_{103}	v_{104}	v_{105}	v_{106}	v_{107}	v_{108}	v_{109}	v_{110}
s_1	1	0	0	1	0	0	0	0	0	0
s_2	0	0	0	0	0	0	1	1	0	1
s_3	1	1	0	0	0	0	0	1	0	0
s_4	0	1	0	1	0	0	0	0	1	0
s_5	0	0	0	0	0	0	0	0	0	0
s_6	0	0	0	0	0	0	0	0	0	0
s_7	0	0	0	0	0	0	0	0	0	0
s_8	0	0	0	0	0	1	0	0	0	0
$S(v_j)$	2	2	0	2	0	1	1	2	1	1

Table 2.8 Continued.

S/V	v_{111}	v_{112}	v_{113}	v_{114}	v_{115}	v_{116}	v_{117}	v_{118}	v_{119}	v_{120}
s_1	1	0	0	0	1	0	0	1	0	0
s_2	0	0	0	0	0	0	0	1	0	0
s_3	1	0	1	0	0	0	0	0	0	0
s_4	0	0	0	1	0	0	1	0	0	0
s_5	0	0	0	0	0	0	0	0	0	0
s_6	0	0	0	0	0	1	0	0	0	0
s_7	0	0	0	0	0	0	0	0	0	0
s_8	0	0	0	0	0	0	0	0	0	1
$S(v_j)$	2	0	1	1	1	1	1	2	0	1

Table 2.8 Continued.

S/V	v_{121}	v_{122}	v_{123}	v_{124}	v_{125}	v_{126}	v_{127}	v_{128}	v_{129}	v_{130}
s_1	0	0	1	0	0	0	0	0	0	0
s_2	1	1	0	1	0	1	1	0	0	1
s_3	1	0	0	1	0	0	0	0	0	0
s_4	1	0	1	0	0	1	1	0	0	0
s_5	0	0	0	0	0	0	0	0	0	0
s_6	0	0	0	0	0	0	0	0	0	0
s_7	0	0	0	0	0	0	1	0	0	0
s_8	0	0	0	0	0	0	0	0	0	0
$S(v_j)$	3	1	2	2	0	2	3	0	0	1

Table 2.8 Continued.

S/V	v_{131}	v_{132}	v_{133}	v_{134}	v_{135}	v_{136}	v_{137}	v_{138}	v_{139}	v_{140}
s_1	0	0	0	0	0	0	1	0	0	0
s_2	1	0	0	0	0	0	0	1	1	1
s_3	0	1	0	1	0	0	0	1	1	0
s_4	1	1	1	0	1	0	1	0	0	1
s_5	0	0	0	0	0	0	0	0	0	0
s_6	0	1	0	0	0	0	1	0	0	0
s_7	0	0	0	0	0	0	0	0	0	0
s_8	1	0	0	0	0	0	0	0	0	0
$S(v_j)$	3	3	1	1	1	0	3	2	2	2

Table 2.8 Continued.

S/V	v_{141}	v_{142}	v_{143}	v_{144}	v_{145}	v_{146}	v_{147}	v_{148}	v_{149}	v_{150}
s_1	1	0	1	0	1	0	0	0	0	0
s_2	0	1	0	0	0	0	0	0	0	0
s_3	0	0	0	0	0	1	0	1	1	0
s_4	1	0	0	0	1	1	1	0	0	0
s_5	0	0	0	0	0	0	0	0	0	0
s_6	0	0	0	0	1	0	0	0	0	0
s_7	0	0	0	0	0	0	0	0	0	0
s_8	0	0	0	0	0	0	0	0	0	0
$S(v_j)$	2	1	1	0	3	2	1	1	1	0

Table 2.8 Continued.

S/V	v_{151}	v_{152}	v_{153}	v_{154}	v_{155}	v_{156}	v_{157}	v_{158}	v_{159}	v_{160}
s_1	1	0	0	0	0	0	0	1	0	0
s_2	0	0	0	0	0	0	0	0	0	0
s_3	1	0	0	0	1	0	1	0	1	0
s_4	0	0	0	0	0	0	0	1	0	1
s_5	0	0	0	0	0	0	0	0	0	0
s_6	0	0	0	0	0	0	0	0	0	0
s_7	0	0	0	0	0	0	0	0	0	0
s_8	0	0	0	0	0	0	0	0	0	0
$S(v_j)$	2	0	0	0	1	0	1	2	1	1

Table 2.8 Continued.

S/V	v_{161}	v_{162}	v_{163}	v_{164}	v_{165}	v_{166}	v_{167}	v_{168}	v_{169}	v_{170}
s_1	0	0	0	0	0	0	1	0	0	0
s_2	0	0	0	0	0	0	0	0	1	0
s_3	0	0	0	0	0	0	1	0	0	0
s_4	1	0	0	0	0	0	0	0	0	0
s_5	0	0	0	0	0	0	0	0	0	0
s_6	0	0	0	0	0	0	0	0	0	0
s_7	0	0	0	0	0	0	0	0	0	0
s_8	0	0	0	0	0	1	0	0	0	0
$S(v_j)$	1	0	0	0	0	1	2	0	1	0

Table 2.8 Continued.

S/V	v_{171}	v_{172}	v_{173}	v_{174}	v_{175}	v_{176}	v_{177}	v_{178}	v_{179}	v_{180}
s_1	0	0	0	1	0	0	0	0	0	0
s_2	0	0	0	0	1	0	0	0	0	1
s_3	1	0	0	0	0	0	1	0	0	0
s_4	0	1	0	0	0	1	0	0	0	0
s_5	0	0	0	0	0	0	0	0	0	0
s_6	0	0	0	0	0	0	0	0	0	0
s_7	0	0	0	1	0	0	0	0	0	0
s_8	0	0	0	0	0	0	0	0	0	0
$S(v_j)$	1	1	0	2	1	1	1	0	0	1

Table 2.8 Continued.

S/V	v_{181}	v_{182}	v_{183}	v_{184}	v_{185}	v_{186}	v_{187}	v_{188}	v_{189}	v_{190}
s_1	0	1	0	0	0	0	1	0	1	1
s_2	0	0	0	0	0	0	0	1	1	1
s_3	0	0	1	1	0	0	0	1	0	0
s_4	0	1	0	1	0	0	1	0	0	0
s_5	0	0	0	0	0	0	0	0	0	0
s_6	0	0	0	0	0	0	0	0	0	0
s_7	0	0	1	0	0	0	0	1	0	0
s_8	0	1	0	0	0	0	0	0	0	0
$S(v_j)$	0	3	2	2	0	0	2	3	2	2

Table 2.8 Continued.

S/V	v_{191}	v_{192}	v_{193}	v_{194}	v_{195}	v_{196}	v_{197}	v_{198}	v_{199}	v_{200}	$f(.)$
s_1	0	0	1	0	0	0	0	0	0	0	38
s_2	0	0	0	0	0	0	0	1	1	0	49
s_3	0	1	0	0	1	1	1	0	1	1	60
s_4	0	0	0	0	1	0	1	1	1	1	62
s_5	0	0	0	0	0	0	0	0	0	0	6
s_6	0	0	0	0	0	0	0	0	0	0	6
s_7	0	0	0	0	0	1	0	0	0	0	6
s_8	0	1	0	0	0	0	0	0	0	0	7
$S(v_j)$	0	2	1	0	2	2	2	2	3	2	234

LIST OF PUBLICATIONS

A: Journals

1. Mohammed, M.A.T., Mohd, W.M.W., Arshah, R.A., Yao, L. (2013). Predefined object reduction. *International Journal of Advanced Research in Computer Engineering & Technology*. 2(12), 3066-3070.

2. Mohammed, M.A.T., Mohd, W.M.W., Arshah, R.A., Yao L. (2013). Object extraction reduction. *International Journal of Science and Advanced Technology*. 3(11), 17-20.

3. Mohammed, M.A.T., Mohd, W.M.W., Arshah, R.A., Yao, L. (2013). Parameter reduction compressions. *Asian Academic Association Research*. 1(19), 686-700.

4. Mohammed, M.A.T., Mohd, W.M.W., Arshah, R.A., Mungad, Sutoyo, E., and Chiroma, H. (2016). Analysis of parameterization value reduction of soft set and its algorithm. *International Journal of Software Engineering & Computer System*. 2, 51-57.

5. Mohammed, M.A.T., & Arshah, R.A. (2016). Fast attribute reduction in soft set. *Asian Academic Association Research*. 3(10), 229-240.

6. Mohammed, M.A.T., & Arshah, R.A. (2016). Soft set reduction based on complement.. *Asian Academic Association Research*. 3(10), 229-240.

B: National and International Conferences

1. Mohammed, M.A.T., Mohd, W.M.W., Arshah, R.A., Mungad, Sutoyo, E., and Chiroma, H. (2015). Complement attributes reduction. *In the proceeding of 2nd International Conference on Advanced Data and Information Engineering (DaEng), Under publishing.*

2. Mohammed, M.A.T., Mohd, W.M.W., Arshah, R.A., Mungad, Sutoyo, E., and Chiroma, H. (2015). Object filtering in soft set. *In the proceeding of 2nd International Conference on Advancd Data and Information Engineering (DaEng), Uner publishing.*

3. Mohammed, M.A.T., Mohd, W.M.W., Arshah, R.A., Mungad, Sutoyo, E., and Chiroma, H. (2015). Hybrid filter for attributes reduction. *In the proceeding of 2nd International Conference on Advanced Data and Information Engineering (DaEng), Under publishing.*

4. Mohammed, M.A.T., Mohd, W.M.W., Arshah, R.A., Mungad, Sutoyo, E., and Chiroma, H. (2015). Hybrid framework parameterization reduction combination. *In the proceeding of 2nd International Conference on Advanced Data and Information Engineering (DaEng), Under publishing.*