AUTOMATIC MULTI-OBJECTIVE CLUSTERING ALGORITHM USING HYBRID PARTICLE SWARM OPTIMIZATION WITH SIMULATED ANNEALING

AHMAD ASAD ABUBAKER

UNIVERSITI SAINS MALAYSIA

2016

AUTOMATIC MULTI-OBJECTIVE CLUSTERING ALGORITHM USING HYBRID PARTICLE SWARM OPTIMIZATION WITH SIMULATED ANNEALING

by

AHMAD ASAD ABUBAKER

**Thesis submitted in fulfilment of the requirements
for the degree of
Doctor of Philosophy**

December 2016

ACKNOWLEDGEMENT

First, I am grateful to Allah for his guidance and protection throughout my life and for giving me courage, patience and strength to carry out this work. This thesis would not have been achieved without reconciling God, then the guidance of my supervisor and field supervisor, along with the support from my parents, wife, brothers, sister, and friends.

I would like to express my deep appreciation and thanks to my supervisor, Prof. Adam Baharum, for the guidance and encouragement and support given to me from the first day that it began a journey toward earning my PhD. His expertise, enthusiasm, constructive criticism and advice strongly aided me to reach my destination. I also would like to thank the field supervisor, Prof. Mahmoud Alrefaei on his efforts, academic and moral support, helpful comments, and valuable advice during the preparation of this work.

Acknowledgement is also worth to all the school, staff, and colleagues at Universiti Sains Malaysia who supported me during my PhD. I would also like to thank Prof. Hailiza Kamarulhaili the dean of our school "School of Mathematical Sciences" who supported me and my colleagues.

I would like to thank my father, mother, brothers, and sister for their love, concern, prayers, and moral support throughout my research. Their encouragement was always a source of motivation for me, and they deserve more thanks than I can give. I also wish to dedicate this thesis to my beloved wife, son, and daughters. I am greatly indebted to their enthusiasm and strong support.

TABLE OF CONTENTS

CHAPTER 4 – EFFICIENCY OF THE PROPOSED ALGORITHM FOR AUTOMATIC CLUSTERING

CHAPTER 5 – SOLVING MULTI-OBJECTIVE OPTIMIZATION PROBLEM

CHAPTER 6 – CONCLUSIONS AND FUTURE WORKS

LIST OF TABLES

LIST OF FIGURES

Page

LIST OF ABBREVIATIONS

AMOSA	Archived Multi-Objective Simulated Annealing
Conn-index	Connectivity-Based Cluster Validity Index
CPSO or CPSOI	Combinatorial Particle Swarm Optimization I
CPSOII	Combinatorial Particle Swarm Optimization II
DB-index	Davies Bouldin Cluster Validity Index
FM	F-measure
GenClustMOO	General Clustering Simulated Annealing Based on Multi-Objective Optimization
GenClustPESA2	General Clustering Pareto Envelope-based Selection Algorithm
KCPSO	K-means with Combinatorial Particle Swarm Optimization
KP	0/1 Knapsack Problem
MDKP	0/1 Multi-Dimension Knapsack Problem
MOCK	Multi-Objective Clustering with Automatic K Determination
MOKP	0/1 Multi-Objective Knapsack Problem
MOMDKP	0/1 Multi-Objective Multi-Dimension Knapsack Problem
MOO	Multi-Objective Optimization Problem
MOPSO	Multi-Objective Particle Swarm Optimization
MOPSOSA	Multi-Objective Particle Swarm Optimization and Simulated Annealing
MOSA	Multi-Objective Simulated Annealing
PSA	Pareto Simulated Annealing
PSO	Particle Swarm Optimization

RNG	Relative Neighborhood Graph
SA	Simulated Annealing
SACLUS	Simulated Annealing Clustering
SAKM	Simulated Annealing with K-means
SF	Sharing Fitness
SL	Single Linkage
Sym-index	Symmetry-Based Cluster Validity Index
VGAPS	Variable String Length Genetic Algorithm Based on Point Symmetry Distance

LIST OF SYMBOLS

P	Dataset
p_i	The i^{th} object in a dataset P
m	The number on objects in a dataset
p_{ij}	The j^{th} feature of i^{th} object
C	Clustering solution
C_i	The i^{th} cluster in the clustering solution C
SN	Stirling number of the second kind
c_i	The center of cluster i
k	The number of clusters
f	Objective function, fitness function, or validity index function
Θ	The feasible solutions set or search space
F	Vector of objective functions or validity indices
f_i	The i^{th} validity index in F
$\mathscr{S}$	The number of objective functions or validity index functions
g_i	The i^{th} inequality constraint in an optimization problem
h_i	The i^{th} equation constraint in an optimization problem
n_{inq}	The number of inequality constraint
n_{eq}	The number of equation constraint
ξ_x	The random variable depend on a solution x
ξ_x^i	The i^{th} sample of a random variable ξ_x
$E(x)$	Estimate of x
$\bar{f}$	Estimate an objective function f
PS	The Pareto optimal set

PF	The Pareto front set
S_i	Measure of intra-cluster distance of cluster i
$d(a,b)$	Euclidean distance between a and b
n_i	The number of objects in cluster i
c_i	The center of cluster i
p_j^i	The j^{th} object in cluster i
DB	Davies-Boulding index value
R_i	Ratio of intra-cluster distance and inter-cluster distance
d_{ps}	Point symmetric distance
p^*	The symmetry point of p
$knear$	The number of nearest neighbor points to p^*
$d_{sym}(p,c)$	The symmetric measure of p with respect to cluster c
$p_{i,j}^*$	The j^{th} point nearest neighbor of the point p_i^*
Sym	Symmetric index value
$lun(x,y)$	The set contains points located within the region of intersection of two circles centered at x and y with radius $d(x,y)$
$d_{short}(x,y)$	Short distance between x and y
$npath$	The number of all paths between two points
ed_j^i	The j^{th} edge in the i^{th} path
ned_i	The number of edges the i^{th} path
$w(ed)$	The edge weight of the edge ed
med_i	The mediod of i^{th} cluster
$Conn$	Connectivity index value
SL_{dis}	The distance between the two closest points of two clusters

CL_{dis}	The distance between the two farthest points of two clusters
AL_{dis}	The average of the distance between all pairs of points of two clusters
DM	The dissimilarity matrix
n	The number of particles
xp_i^t	The best previous position of the i^{th} particle during iteration 1 to t
xg^t	The best position among all the particles in the swarm during iteration 1 to t
x_i^t	The position of the i^{th} particle at iteration t
v_i^t	The velocity for particle i at iteration t
w	The initial weight
r_1^t, r_2^t	Random number in $[0,1]$
v_{min}	Minimum velocity
v_{max}	Maximum velocity
k_i^t	The best number of clusters for particle i at iteration t
k_{min}	Minimum number of cluster
k_{max}	Maximum number of cluster
$k_{i,pbest}^t$	The best number of cluster of particle i during iteration 1 to t
k_{gbest}^t	The best number of cluster of all particles during iteration 1 to t
$N(x)$	The set of neighbors of x
$R(x,\acute{x})$	The probability of selecting $\acute{x}$ from $N(x)$
T_0	The initial temperature
T_{min}	The final temperature

$g(T_t,t)$	Decrement rate
T_t	Temperature at iteration t
$\mathscr{L}_t$	Increasing sequence of positive integer
$V_t(x)$	The number of times that algorithm visited solution x in first t iterations
x_t^*	The estimate optimal solution after t iterations
$iter$	The number of iterations
$\Delta dom_{a,b}$	The acceptance probability of a new solution b where a is the current solution
HL	The maximum size of the archive
SL	The maximum size to which the archive may be filled before clustering is used to reduce its size to HL
$\hat{k}$	The number of subcluster
a_i	The i^{th} solution in the archive
$\bar{c}_j^i$	The center of the j^{th} sub-cluster of the i^{th} cluster
X_i^t	The vector with m components that represent the position of the particle i at iteration t
X_{ij}^t	The position component that represent the cluster number of the j^{th} object in the i^{th} particle
V_i^t	The vector with m components that represent the velocity of the particle i at iteration t
V_{ij}^t	The velocity component that represent the motion of the j^{th} object in the i^{th} particle

XP_i^t	The vector with m components that represent the best previous position of the particle i at iteration t
XP_{ij}^t	The j^{th} component of the XP_i^t at iteration t
XG_i^t	The vector with m components that represent the leader position of the particle i at iteration t
XG_{ij}^t	The j^{th} component of the XG_i^t at iteration t
$Xnew_i$	The candidate clustering solution of the particle i
$Vnew_i$	The candidate velocity of the particle i
X_i^{MOSA}	The clustering solution of the particle i obtained from MOSA
V_i^{MOSA}	The velocity of the particle i obtained from MOSA
SR	The size of the repository
CR^t	The set of the current solutions in the repository at iteration t
$NDXP^t$	The set of no-dominated solution from all XP_i^t, $i = 1,\ldots,n$
$CRNDXP^t$	The set of no-dominated solution from two sets CR^t and $NDXP^t$
$\mathscr{X}_i$	The i^{th} solution in $CRNDXP^t$
$fshar(\mathscr{X}_i)$	The fitness sharing for $\mathscr{X}_i$
nc_i	The niche count for $\mathscr{X}_i$
$sharing_i^j$	The measure of similarity between $\mathscr{X}_i$ and $\mathscr{X}_j$
σ_{share}	The distance to keep solution a way from each other
$Sim(C_j, \hat{C}_k)$	The similarity function between two cluster C_j and $\hat{C}_k$, which known as Jaccard coefficient
W, R_1 and R_2	The vectors of m component with value 0 or 1
w, r_1 and r_2	The probability to generate W, R_1 and R_2 respectively

$MS(T,C)$	The Minkowski score between actual solution T and selected solution C
$P(T,C)$	The precision, which is the ratio of the points in cluster C that exist in class T
$R(T,C)$	The recall, which is the ratio of the points in class T that exist in cluster T
$F(T,C)$	F-measure that provides the value of similarity between T and C
k_T and k_C	The number of clusters in T and C respectively
BVF	The best values of F-measure
MVF	The maximum value of F-measure
OC	Ordering cost
ST	Setup cost
KI	The number of items that are ordered
lt	The lead time required to reach the order from the supply to the company
(s,S)	The policy to re-inventory level to amount S if the level drops below the number s
L	Inventory level
$L(t)$	Inventory level at time t
D	The size of items that demand
β	The cost one item
pro_j	Probability to demand j items
MS	The maximum size of items that allowed to demand

HC	Holding cost
SC	Shortage cost
AHC	The average holding cost per month
ASC	The average shortage cost per month
AOC	The average ordering cost per month
$\bar{L}^{+}$	The average number of items exist in the inventory per month
$L^{+}(t)$	The number of items exist in the inventory at time t
$\bar{L}^{-}$	The average number of items in the backlog per month
$L^{-}(t)$	The number of items in the backlog at time t
$\mathscr{M}$	The number of months
$\mathfrak{n}$	The number of items
$\mathfrak{p}_i$	The profit of item i
$\mathfrak{w}_i$	The weight of item i
$\mathfrak{c}$	The capacity of knapsack
$\mathscr{N}$	The set of items
ω	The binary solution
ω_i	The binary value of item i
$\mathfrak{d}$	The number of constraints in NDKP
$\mathfrak{w}_{ij}$	The weight of item i in constraint j
$\mathfrak{p}_{ij}$	The profit of item i in objective function j
$\mathfrak{c}_j$	The capacity of constraint j
NNDS	The number of non-dominant solution
NI	The number of iteration to obtain all non-dominant solution

ALGORITMA KELOMPOK MULTI OBJEKTIF AUTOMATIK MENGGUNAKAN PENGOPTIMUMAN PARTIKEL SEKAWAN HIBRID DENGAN SIMULASI PENYEPUHLINDAPAN

ABSTRAK

Pengelompokan adalah suatu teknik pelombongan data. Di dalam bidang set data tanpa selia, tugas mengelompok ialah dengan mengumpul set data kepada kelompok yang bermakna. Pengelompokan digunakan sebagai teknik penyelesaian di dalam pelbagai bidang dengan membahagikan dan mengstruktur semula data yang besar dan kompleks supaya menjadi lebih bererti justru mengubahnya kepada maklumat yang berguna. Di dalam tesis ini, satu teknik automatik baru berdasarkan pengoptimuman kawanan zarah pelbagai objektif dan penyepuhlindapan bersimulasi (MOPSOSA) diperkenalkan. Algoritma yang dicadangkan mampu menjalankan pengelompokan automatik yang tepat untuk pembahagian dataset ke dalam bilangan kelompok yang sesuai. MOPSOSA menggabungkan ciri-ciri kaedah K-means, pengoptimuman kawanan zarah pelbagai objektif, penyepuhlindapan bersimulasi pelbagai objektif, dan teknik berkongsi kecergasan. Tiga indeks kesahihan kelompok telah dioptimumkan serentak untuk mewujudkan bilangan kelompak yang sesuai dan pengelompokan yang tepat untuk sesuatu set data. Indeks kesahihan kelompok pertama adalah berdasarkan jarak Euclid, indeks kesahihan kelompok kedua adalah berdasarkan kepada jarak titik simetri, dan indeks kesahihan kelompok terakhir adalah berdasarkan jarak pendek. Tiga algoritma pengelompokan objektif tunggal dan tiga algoritma pengelompokan automatik pelbagai objektif telah dibandingkan dengan algoritma MOPSOSA dalam menyelesaikan masalah pengelompokan dengan menentukan bilangan kelompok yang sebenar dan pengelompokan optimum. Ujikaji pengiraan telah dijalankan untuk mengkaji empat belas set data buatan dan lima set data sebenar. Hasil ujikaji pengiraan menunjukkan bahawa algoritma MOPSOSA yang dicadangkan memperolehi ketepatan pengelompokan yang lebih baik berbanding dengan algoritma lain. Selain itu, kecekapan algoritma MOPSOSA dikaji berdasarkan perubahan dalam kebarangkalain parameter halaju zarah. Sembilan belas set data buatan dan sebenar digunakan untuk menggambarkan kesan parameter halaju ke atas kecekapan algoritma MOPSOSA. Keputusan menunjukkan bahawa kecekapan algoritma MOPSOSA boleh ditingkatkan dengan meninggikan nilai parameter kebarangkalian halaju. Keadaan ini benar hingga ke suatu nilai tertentu, yang selepas itu kesan positif meninggikan parameter kebarangkalian halaju akan sebaliknya menjadi kesan negatif. Akibatnya, nilai sesuai parameter kebarangkalian halaju dapat ditentukan. Tambahan pula, satu prosedur untuk menyelesaikan masalah pengoptimuman pelbagai objektif dengan menjumlahkan kedua-dua algoritma tersebut, iaitu penyepuhlindapan bersimulasi pelbagai objektif (MOSA) dan MOPSOSA dicadangkan. Prosedur ini digunakan untuk menyelesaikan dua masalah pengoptimuman pelbagai objektif yang praktikal, iaitu masalah sistem inventori pelbagai objektif dan beg galas 0/1 pelbagai objektif bermultidimensi. Suatu set penyelesaian yang kecil diperolehi dengan menggunakan MOSA+MOPSOSA dan sebaliknya bukan sebilangan besar penyelesaian dalam set Pareto yang dengan itu, membolehkan pembuat keputusan memilih penyelesaian yang betul dengan mudah. Untuk meningkatkan prosedur ini, empat jadual penyejukan yang berbeza, iaitu, tetap, eksponen, linear dan logaritma dibincangkan dan dibandingkan antara satu sama lain dalam algoritma MOSA. Perbandingan keputusan menunjukkan bahawa jadual penyejukan tetap adalah lebih baik daripada yang lain. Oleh itu, jadual penyejukan ini

digunakan dalam prosedur yang dicadangkan.

AUTOMATIC MULTI-OBJECTIVE CLUSTERING ALGORITHM USING HYBRID PARTICLE SWARM OPTIMIZATION WITH SIMULATED ANNEALING

ABSTRACT

Clustering is a data mining technique. In the field of unsupervised datasets, the task of clustering is by grouping the dataset into meaningful clusters. Clustering is used as a data solution technique in various fields to divide and restructure the large and complex data to become more significant thus transform them into useful information. In this thesis, a new automatic clustering algorithm based on multi-objective particle swarm optimization and simulated annealing (MOPSOSA) was introduced. The proposed algorithm is capable of automatic clustering, which is appropriate for partitioning datasets into a suitable number of clusters. MOPSOSA combines the features of K-means method, multi-objective particle swarm optimization, multi-objective simulated annealing, and sharing fitness technique. Three cluster validity indices were optimized simultaneously to establish the suitable number of clusters and the appropriate clustering for a dataset. The first cluster validity index is based on Euclidean distance, the second cluster validity index is based on point symmetry distance, and the last cluster validity index is based on short distance. Three single-objective clustering algorithms and three multi-objective automatic clustering algorithms have been compared with the MOPSOSA algorithm in solving clustering problems by determining the actual number of clusters and optimal clustering. Computational experiments were conducted to study fourteen artificial and five real-life datasets. Computational experimental result shows that the proposed MOPSOSA algorithm obtained better clustering accuracy compared with the other algorithms. Moreover, the efficiency of the MOPSOSA algorithm is studied on the basis of the change in the probability velocity parameters of particles. Nineteen artificial and real-life datasets are used to illustrate the effect of velocity parameters on the efficiency of the MOPSOSA algorithm. The results show that the efficiency of the MOPSOSA algorithm may be enhanced by raising the probability velocity parameters values. This is true up to a specific value, after which, the positive effect of increasing the probability velocity parameters becomes a negative effect, instead. Consequently, the suitable values of probability velocity parameters have been identified. Furthermore, a procedure for solving multi-objective optimization problems by aggregating the two algorithms, that is the multi-objective simulated annealing (MOSA) and MOPSOSA were proposed. This procedure is used to solve two practical multi-objective optimization problems, namely, the multi-objective inventory system and the 0/1 multi-objective multi-dimension knapsack problems. A small set of solutions is obtained using MOSA+MOPSOSA instead of a large number of solutions in the Pareto set, thereby allowing a decision maker to select a proper solution easily. To improve this procedure, four different cooling schedules, namely, constant, exponential, linear, and logarithmic, are discussed and compared with each other in the MOSA algorithm. Comparison results show that the constant cooling schedule is better than the others. Thus, this cooling schedule is used in the proposed procedure.

CHAPTER 1

INTRODUCTION

The development of science and their applications in various fields have contributed to an increase in the amount and diversity of data. The data that can be collected from various fields have no benefit unless sound analysis is conducted to obtain valuable information. Thus, such data have to be classified, summarized, and understood. Data mining transforms a large collection of data into knowledge (Han et al., 2011). In this thesis, the focus is on clustering, one of the important techniques in data mining.

1.1 Overview on Clustering

Clustering (Kaufman and Rousseeuw, 2009) is a data mining technique in the field of unsupervised datasets; this technique is used to explore and understand large collections of data. In clustering unsupervised datasets, the structural characteristics of data are unknown and unlabeled. Given a dataset P of m objects, the task in clustering process is grouping the dataset into k meaningful groups called clusters.

The clustering has widespread applications in many fields such as the following:

- **Gene expression data**: Clustering is an effective technique to discover clusters of similar objects in gene expression data so that biologists can identify potentially meaningful connections among those objects (Eisen et al., 1998; Hughes et al., 2000; Yeung et al., 2003).

- **Marketing**: In market research, clustering has been used to divide the market into homogeneous clusters of customers with similar behavior, and to use the resulting information in developing targeted marketing (Christopher, 1969; Saunders, 1980; Kuo et al., 2002).

- **Image segmentation**: Image segmentation is the task of subdividing an image into different regions of certain properties and extracting the desired parts. Clustering is used to detect borders of regions in an image (Coleman and Andrews, 1979; Cai et al., 2007; Wang and Pan, 2014).

1.2 Concepts of Clustering

This section, presents certain concepts and notations that are frequently used in the literature on clustering.

- **Object (pattern, sample, data point, observation, item, or individual):** Object p is a single datum in dataset $P = \{p_1, p_2, \ldots, p_m\}$, where m is the number of objects in the dataset. The i^{th} object $p_i = \{p_{i1}, p_{i2}, \ldots, p_{id}\}$ consists of a vector of $d-$dimension (Gan et al., 2007).

- **Feature (attribute or variable):** Feature p_{ij} is the j^{th} individual scalar component of the object i (Gan et al., 2007).

- **Cluster (or group):** A cluster is a collection of data objects with features that are similar to one another, and dissimilar features to objects in other clusters. (Jain and Dubes, 1988).

- **Validity index (or cluster validity):** Validity index is a measure that is used to evaluate the results of clustering (Gan et al., 2007).

- **Distance and similarity measure:** Distance and similarity are used to quantitatively describe the similarity or dissimilarity between two objects, objects with clusters, or two clusters (Jain and Dubes, 1988).

1.3 Problem Statement

The clustering of dataset that contains objects is the distribution of these objects into proper number of clusters that contain objects having the same features.

1.3.1 Clustering Problem

The clustering problem can be defined as follows (Masoud et al., 2013): Consider a dataset $P = \{p_1, p_2, \ldots, p_m\}$ with m objects. The clustering of dataset P is the distribution of objects that exist in P into k clusters $C = \{C_1, C_2, \ldots, C_k\}$, where C is called a clustering solution, and C_i is the i^{th} cluster in C, such that the following properties are satisfied:

- $$\bigcup_{i=1}^{k} C_i = P, \tag{1.1}$$

- $$C_i \bigcap C_j = \phi, \;\; i \neq j, \;\; i = 1, \ldots, k, \;\; j = 1, \ldots, k, \tag{1.2}$$

- $$C_i \neq \phi, \;\; i = 1, \ldots, k. \tag{1.3}$$

Stirling numbers of the second kind $SN(m,k)$ (Pak, 2005) are used to calculate the number of possible ways to divide a dataset of m objects into k non-empty clusters (number of feasible solutions), where $SN(m,k) = \frac{1}{k!} \sum_{i=1}^{k} (-1)^{k-i} \binom{k}{i} (i)^m$. For example, let the number of objects $m = 150$, and the number of clusters $k = 3$, then there are more than 6×10^{70} different solutions. Although the cluster numbers of the previous example are known, the number of solutions is large. Thus, the clustering problem can be structured as a single or multi-objective optimization problem.

1.3.2 Single-Objective Function for the Clustering Problem

The clustering optimization problem can be formulated as the following single-objective function:

$$\begin{aligned} &\underset{C \in \Theta}{\text{minimize}} \quad f(C) \\ &\text{subject to} \quad C \ \text{ satisfies the constraints (1.1, 1.2, and 1.3)} \end{aligned} \tag{1.4}$$

where f is the validity index function, Θ is the feasible solutions set that contains all possible clustering solutions for the dataset P of m objects into k clusters and $C = \{C_1, C_2, \dots, C_k\}$ is a vector of k clusters, $k = 2, 3, \dots, m-1$. The optimal solution is given by; $C^* \in \Theta$ such that $f(C^*) = \min\{f(C) \mid C \in \Theta\}$.

1.3.3 Multi-Objective Function for Clustering Problem

The single evaluation function is often ineligible to determine the appropriate clusters for a dataset; thus, it provides an inferior solution (Suresh et al., 2009). Accordingly, the clustering problem is structured as a multi-objective optimization problem where different validity indices can be applied and evaluated simultaneously.

The multi-objective clustering problem for $\mathscr{S}$ different validity indices is defined as follows:

$$\begin{aligned} \underset{C\in\Theta}{\text{minimize}} \quad & F(C) = [f_1(C), f_2(C), \ldots, f_{\mathscr{S}}(C)] \\ \text{subject to} \quad & C \text{ satisfied the constraints (1.1, 1.2, and 1.3)} \end{aligned} \tag{1.5}$$

where F is a vector of $\mathscr{S}$ validity indices, and f_i is the i^{th} validity index in F. There may be no solution that minimize all the $f_i(C)$ validity indices. Therefore, the aim is to construct the Pareto optimal set. The Pareto set contains all solutions in which cannot find any solution in the search space dominate them. This solution is called non-dominant solution. Further information on the Pareto optimal set, is provided in Section 2.2.2.

1.4 Research Motivation and Research Questions

The expansion of datasets has led to larger and more complex data with no structure, significance, and substance. It is difficult to understand data with such situation. Clustering is used as a data solution technique in various fields to divide and restructure the data to become more significative and to transform them into useful information. Currently, clustering is a difficult problem. This is due to the appropriate number of clusters is unknown, the large number of potential solutions, and the dataset being unsupervised. To solve this problem, the number of clusters that fits a dataset must be determined, and the objects for these clusters must be assigned appropriately. Therefore, dealing with various shapes and sizes of datasets without providing the proper clustering or knowing the cluster number is a challenge.

The main motivation for this work is to improve the effectiveness of clustering a dataset with different sizes, shapes, dimensions, overlapping, convex and non-convex datasets, as well as unknown numbers of clusters. Clustering the dataset, can provide insights into these datasets and an improved understanding of their characteristics.

The present study focuses the following main questions:

1. How to conduct proper clustering with high accuracy for dataset with various shapes, sizes, and dimensions as well as for overlapping dataset?

2. How to detect the suitable number of clusters for any dataset?

3. How to solve the clustering problem in fast convergence and prevent stagnation in local solutions?

4. How to help decision makers in choosing a suitable solution from among a large number of overlapping solutions in the Pareto set?

1.5 Research Objectives

Several automatic clustering algorithms that have been proposed in previous studies can be used to solve the clustering problems and are highly important in many applications. Although the clustering of a dataset is the main objective of the present study, it is insufficient. Achieving the target to detect the appropriate number of clusters and proper partition of various datasets in these clusters with high accuracy is the most important target. The primary objectives of this study can be summarized by the following:

- To develop a new automatic clustering algorithm based on multi-objective optimization, namely, hybrid multi-objective particle swarm optimization with simulated annealing (MOPSOSA).

- To determine the efficiency of the new automatic clustering algorithm based on the changes in the velocity parameters of particle swarm optimization.
- To determine the efficiency of the multi-objective simulated annealing (MOSA) based on the types of cooling schedules.
- To compare the performance of the proposed algorithm with the performances of six clustering techniques.
- To minimize/optimize the number of solutions in the Pareto set by clustering the Pareto set into clusters containing similar feature solutions.

1.6 Research Contributions

Simulated Annealing (SA) requires more computational time than does particle swarm optimization (PSO) (Shieh et al., 2011). The former requires low variations of temperature parameters to obtain a global solution (Mitra et al., 1985). Some of the particles may become stagnant and remain unchanged, especially when the objective functions of the best personal position and the best global position are similar (Shieh et al., 2011). Thus, the particle cannot jump out, which in turn causes convergence toward the local solution and the loss of its ability to search for the optimal Pareto set. This phenomenon is a disadvantage compared with SA, which can jump away from a local solution.

The main approach that has led to the accuracy of the proposed MOPSOSA algorithm in solving the clustering problem is merging the advantages of fast calculation and convergence in PSO with the ability to evade local solutions in SA. This merge is achieved by developing combinatorial PSO to a multi-objective particle swarm optimization (MOPSO) to simultaneously address three different cluster validity indices. Additionally, This study has successfully delivered the effect of the velocity parameters that controls the movement of particles in the efficiency of the MOPSOSA algorithm.

In solving multi-objective optimization problems, choose a proper solution from among a large number of Pareto solutions is a challenge for a decision maker. This study helps decision makers in choosing a suitable solution from among a large number of overlapping and complex Pareto solutions in two real life problems, namely, multi-objective inventory system and 0/1 multi-objective multi-dimension knapsack problem.

1.7 Methodology

Several algorithms are proposed to optimize the clustering of a dataset based on single or multi cluster validity indices. Some of these algorithms require the actual number of clusters, and others estimate the suitable number of clusters. Many of the proposed algorithms have been developed that used only one technique to solve the clustering problem rather than merging more than one techniques. This thesis integrates four techniques into the proposed algorithm to improve its performance and to obtain high accuracy in solving the clustering problems. This approach involves a combination of K-means, MOPSO, sharing fitness (SF), and MOSA techniques. The research framework is illustrated in Figure 1.1.

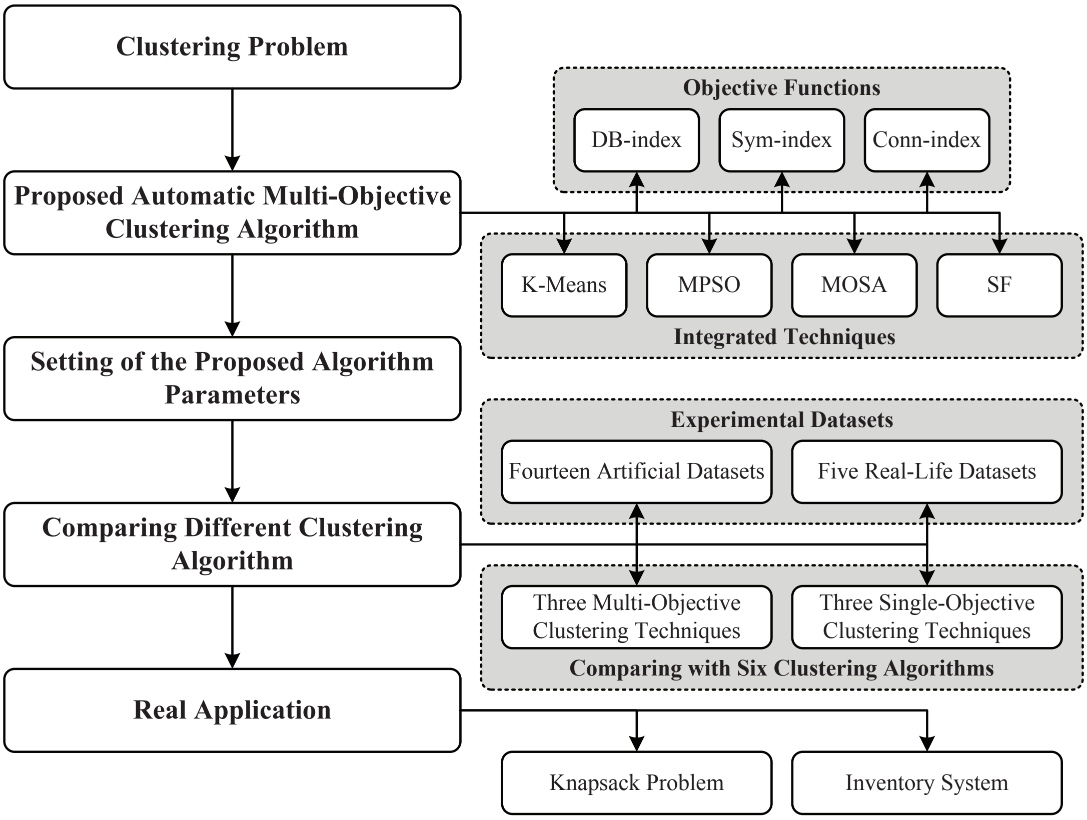

Clustering Problem
Objective Functions
DB-index
Sym-index
Conn-index
Proposed Automatic Multi-Objective Clustering Algorithm
K-Means
MPSO
MOSA
SF
Integrated Techniques
Setting of the Proposed Algorithm Parameters
Experimental Datasets
Fourteen Artificial Datasets
Five Real-Life Datasets
Comparing Different Clustering Algorithm
Three Multi-Objective Clustering Techniques
Three Single-Objective Clustering Techniques
Comparing with Six Clustering Algorithms
Real Application
Knapsack Problem
Inventory System

Figure 1.1: Flowchart of the research process

Initially, the K-means method is used to improve the selection of the initial *n* particle position because of its significance in the overall performance of the search process. The position that a particle signifies is a candidate solution to the optimization problem. Then the PSO technique uses the *n* initial particle position to search the Pareto optimal solutions through the feasible solution set, where each particle seeks a better position in the search space. A performance assessment of each particle is conducted according to three different cluster validity indices simultaneously. The first validity index is Davies-Bouldin index is called *DB*-index (Davies and Bouldin, 1979), which is based on Euclidean distance; the second is symmetry-based cluster validity indices called *Sym*-index (Bandyopadhyay and Saha, 2008), which is based on point symmetry distance; and the last is a connectivity-based cluster validity index called *Conn*-index (Saha and Bandyopadhyay, 2012), which is based on short distance. If no change occurs in a particle position or when the particle moves to a bad position, then MOSA is used to improve the particle search. The proposed algorithm may generates a large number of Pareto optimal solutions through a trade-off among the three different validity indices. Therefore, SF (Goldberg and Richardson, 1987) is used to maintain diversity in the repository that contains Pareto optimal solutions.

The efficiency of the proposed algorithm under various parameters are studied by applying the algorithm on 19 datasets. This step is followed by studying the efficiency of the proposed algorithm through comparison with the performance of six clustering algorithms, three automatic multi-objective clustering techniques, and three single-objective clustering techniques.

Then, propose a procedure to solve two multi-objective optimization problems, namely, multi-objective inventory system and 0/1 multi-objective multi dimension knapsack problem by construct a small set of the solution instead of a large number of solutions in the Pareto set, which assist decision-maker in choosing an appropriate solution. The proposed procedure is divided into two main stages; the first stage is to obtain a Pareto set, and the second stage is to prune Pareto set.

In this thesis, Matlab is used as programming language in the numerical examples and was run using a computer model HP Envy desktop (Intel Core i7-4790, CPU 3.60 GHz, 16.0 GB, 2 TB, 64-bit OS Windows 8.1).

1.8 Thesis Organization

The thesis consists of six chapters that are organized as follows:

Chapter 1 provides an overview of clustering and some basic concepts, followed by the definition of the single and multi-objective clustering problems. The motivation, objectives, and contributions of the study are also summarized. Additionally, the research methodology is described.

Chapter 2 presents the important concepts related to this study. This chapter is divided into four main parts. In the first part, a single objective optimization problem, multi-objective optimization problem, the concept of non-dominant solution, and Pareto set are presented. The second part explains the encoding scheme of the clustering solution. In the third part, a number of cluster validity indices are described. The fourth part explains the ideas for certain clustering techniques, namely, K-means, single-linkage, clustering by PSO, and clustering by SA.

Chapter 3 presents in details the proposed automatic clustering multi-objective algorithm that used to solve the clustering problem. Then, 19 datasets are used in the experiment to measure the clustering quality. This chapter also compares the proposed algorithm with three automatic multi-objective clustering techniques and three single-objective clustering techniques.

Chapter 4 discusses the efficiency of the new proposed algorithm based on the change in the three important probability velocity parameters that control the movement of particles.

Chapter 5 presents the procedure to solve two important multi-objective optimization problems, namely, multi-objective inventory system and multi-objective multi-dimension 0/1 knapsack problems. This procedure uses MOSA and the proposed algorithm. Additionally, the effects of several types of cooling schedules on the MOSA algorithm are discussed and compared.

Chapter 6 summarizes the conclusions, and contributions of this study as well as presents suggestions for further research.

CHAPTER 2

LITERATURE REVIEW

2.1 Introduction

To solve the clustering problem, we consider the detecting solutions, whose validity indices show the highest accuracies. Furthermore, as the number of clusters may not be known, finding the best solution to the clustering problem becomes more difficult. These solutions are defined as the optimal solutions set, which consists of the non-dominant solutions in the feasible solutions set (search space). These non-dominant solutions are simultaneously obtained from the search space, based on the minimized multi validity index functions. In general, the number of feasible solutions in the search space for clustering problem is huge.

In this chapter, single-objective optimization problem, multi-objective optimization problem, the concept of non-dominant solution, and the Pareto set are discussed. Various algorithms to solve clustering problems are discussed. These include K-means, single-linkage, clustering by PSO, MOPSO, SA, and MOSA algorithms. Some of these algorithms are based on multi validity indices, but most rely exclusively on one validity index. Furthermore, for some of these algorithms, the number of clusters must be known. The performances of all these algorithms are also affected depending on the size and shape dataset.

The clustering technique seeks to distribute a dataset into clusters of similar features. Evaluating the goodness of clustering solutions resulting from the clustering algorithms is important. Therefore, in this chapter, we present three important validity indices with corresponding different distances, namely, *DB*-index, *Sym*-index, and *Conn*-index.

2.2 Optimization Problem

At the heart of any decision, a deterministic or stochastic optimization problem can be found. Furthermore, the optimization problem deals with maximizing or minimizing single or multi-objective functions. Usually, the number of feasible solutions in the search space is very large, and the aim is to detect one or more optimal solutions for these objective functions.

2.2.1 Single-Objective Optimization Problem

The single-objective optimization problem revolves around choosing a solution from a search space to optimize a certain targeted objective. Without loss of generality, the general single-objective optimization problem can be represented mathematically as the following minimization problem:

$$\begin{aligned} \min_{x \in \Theta} \quad & f(x) \\ \text{subject to} \quad & g_i(x) \leq 0, \ i = 1, 2, \ldots, n_{inq}. \\ & h_j(x) = 0, \ j = 1, 2, \ldots, n_{eq}. \end{aligned} \tag{2.1}$$

where Θ is the search space that contains all potential solution candidates or the feasible solutions. In this thesis, we consider the search space contains a huge finite solutions that is defined as follows: $\{x \mid g_i(x) \leq 0$ and $h_j(x) = 0, \forall\, i = 1, 2, \ldots, n_{inq}$ and $j = 1, 2, \ldots, n_{eq}\}$, where x is a solution, $g_i(x)$ is the i^{th} inequality constraint, $h_j(x)$ is the j^{th} equality constraint, n_{inq} and n_{eq} are the number of inequality and equality constraints, respectively, and f is the objective function or the expected objective function of a complex deterministic or stochastic system, respectively. In a stochastic optimization problem, due to the stochastic nature of f, the optimization problem 2.1 is expressed as follows:

$$\min_{x \in \Theta} f(x) \equiv E[SP(x, \xi_x)] \tag{2.2}$$

where the sample performance SP is a deterministic function of x and ξ_x, in which ξ_x is a random variable depending on the x. A stochastic optimization technique is used for solving the optimization problem 2.2, where the objective function values are estimated using simulation (i.e. random generation of samples of stochastic process $\xi_x^1, \xi_x^2, \ldots, \xi_x^t$ of the random variable ξ_x). $E[SP(x, \xi_x)]$ is estimated by sampling, to obtain the best estimated solution over Θ. This is expressed as follows:

$$E[SP(x, \xi_x)] \approx \frac{1}{t} \sum_{i=1}^{t} SP(x, \xi_x^i) \equiv \bar{f}(x) \tag{2.3}$$

where $\bar{f}(x)$ is the estimated performance of $f(x)$, ξ_x^i represents the i^{th} sample randomness of solution x, and t is the number of replications (i.e., the number of simulation runs).

2.2.2 Multi-Objective Optimization Problem

Many real-life problems are considered Multi-Objective Optimization problems (MOO), which contain several objectives that must be optimized simultaneously. Without loss of generality, the minimization of MOO can be expressed as follows:

$$\begin{aligned} \min_{x\in\Theta} \quad & F(x) = (f_1(x), f_2(x), \ldots, f_{\mathscr{S}}(x)) \\ \text{subject to} \quad & g_i(x) \leq 0,\ i = 1, 2, \ldots, n_{inq}. \\ & h_j(x) = 0,\ j = 1, 2, \ldots, n_{eq}. \end{aligned} \tag{2.4}$$

where $f_k(x)$ is the k^{th} objective function, and $F(x)$ is a vector of $\mathscr{S}$ objective functions.

Many algorithms have been proposed to solve MOO by detecting a set of non-dominant solutions called the Pareto optimal set (PS), which is defined in Definition 2.2.2. The non-dominant concept, Pareto set and Pareto front set (PF), are defined in the following definitions.

Definition 2.2.1 (Pareto Dominance) *Consider x and $\hat{x}$ as two solutions in the feasible solutions set Θ. The solution x is said to be dominated by the solution $\hat{x}$ if and only if $f_i(\hat{x}) \leq f_i(x)$, $\forall\ i = 1, \ldots, \mathscr{S}$ and $f_i(\hat{x}) < f_i(x)$ for at least one i, and denoted by $\hat{x} \prec x$. Otherwise, x is said to be non-dominated by $\hat{x}$, and denoted by $\hat{x} \nprec x$.*

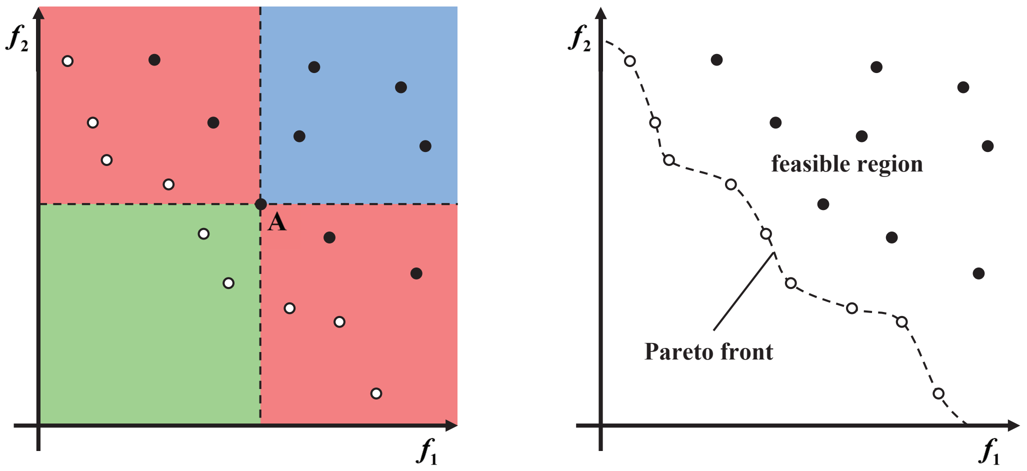
Definition 2.2.2 (Pareto Optimal Set) *PS is a set that includes all non-dominated solutions in the feasible solutions set Θ. PS is defined as follows:*

$$PS = \{\, x \in \Theta \mid\ \hat{x} \nprec x\,,\ \forall\, \hat{x} \in \Theta \,\}. \tag{2.5}$$

Definition 2.2.3 (Pareto Front Set) *For a given PS for MOO, PF of the objective functions $F(x) = (f_1(x), f_2(x), \ldots, f_{\mathscr{S}}(x))$ is defined as follows:*

$$PF = \{\, F(x) \mid\ x \in PS \,\}. \tag{2.6}$$

Figure 2.1a explains the dominance relationship between solutions of minimized system of two objective functions f_1 and f_2. Suppose the reference solution is solution A. There are three different regions of dominance relations related to solution A. Solution A dominates all the solutions located in blue region, because the solution is better in both two objectives f_1 and f_2. On the other hand, all solutions located in the green region dominate solution A. The solutions that are located in the red regions, are neither dominant nor being dominated by solution A. Figure 2.1b shows the PF that contains the non-dominated solutions represented by white circles; as can be seen, no solution in the feasible solutions set dominates them.



(a) Dominance relationship between solution A and other solutions.

(b) White circles represent non-dominant solutions of the Pareto front set.

Figure 2.1: Dominance relation and PF of the minimization system of two objective functions f_1 and f_2.

Fonseca and Fleming (1995) are the first to use the idea of Pareto optimality, to solve MOO by detecting the non-dominated solutions. To enrich the theory in this field. Ulungu et al. (1995), for example, proposed the multi-objective simulated annealing algorithm to solve multi-objective combinatorial optimization problems by finding the

Pareto set of solutions. Zitzler and Thiele (1999) proposed the strength Pareto evolutionary algorithm, which, starts as an empty set called an archive. In each iteration, all non-dominated sets in the population are copied within the archive set and then updated to remove any dominated individual or duplicate. Lee et al. (2004) attempted to solve MOO problem by incorporating the concept of Pareto optimality into the ranking and selection scheme; they proposed the multi-objective optimal computing budget allocation technique to identify all non-dominated designs by allocating simulation replications to the designs. Alrefaei and Diabat (2009) proposed two algorithms, which are based on the idea of simulated annealing with constant temperature to solve MOO problems.

2.3 Integer Encoding Scheme

A number of encoding schemes of clustering solutions have been proposed in the literature. For example, Hruschka et al. (2009) categorized the encoding schemes into three types: binary, integer and real. The clustering solution in the integer encoding scheme is an integer vector of m labels. The i^{th} component represents the cluster number of the object i that has a value between 1 and k, where k is the number of clusters. In fact, there are $k!$ different forms of vector that represent the same solution. Figure 2.2 shows example of 3! redundant solutions that represent the same clustering solution for clustering 9 objects into 3 clusters, namely, [111332222], [111223333], [222113333], [222331111], [333112222], and [333221111]. Such a case can be addressed using the re-numbering procedure (Falkenauer (1998)).

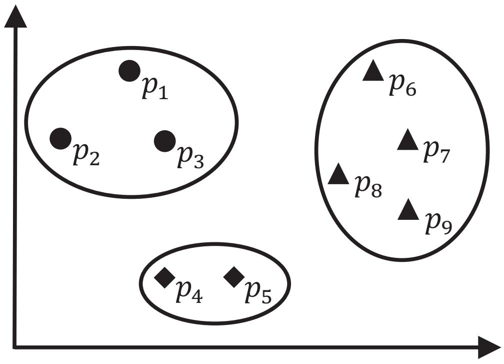


Figure 2.2: An example for clustering 9 objects into 3 clusters.

2.4 The Cluster Validation

In this section, a number of cluster validity indices proposed in the literature are discussed to validate the clustering solutions related to automatic clustering. The validity index study is essential in order to detect the appropriate clustering and proper number of clusters for a dataset. One of the important cluster validity indices is *DB*-index (Davies and Bouldin, 1979), which has been adopted by Bandyopadhyay and Maulik (2001), Bandyopadhyay and Maulik (2002), Lai (2005), Liu et al. (2011), and Masoud et al. (2013) to measure the validity clustering. The *Sym*-index (Bandyopadhyay and Saha, 2008) that was developed to be able to determine any kind of symmetric cluster from dataset. Bandyopadhyay and Saha (2008) proposed the *Sym*-index, which utilizes the point symmetry distance developed by Bandyopadhyay and Pal (2007). The *Sym*-index inspired by the *I*-index that proposed by Maulik and Bandyopadhyay (2002). Instead of the Euclidean distance and the point symmetry distance, a new measure of connectivity called *Conn*-index has recently been incorporated in the definitions of the seven cluster validity indices, (Saha and Bandyopadhyay (2012)). The validity indices, namely, *DB*-index, *Sym*-index and *Conn*-index, are further described below.