A Major Project Report On

FUZZY CLUSTERING USING GRAVITATIONAL SEARCH ALGORITHM

Submitted in partial fulfilment of the requirements

for the award of the degree of

MASTER OF TECHNOLOGY

IN

SOFTWARE ENGINEERING

By

Himanshu Kaul

(Roll No. 2K13/SWE/04)

Under the guidance of

Prof. OP Verma

Professor

Department of Computer Engineering

Delhi Technological University, Delhi

Department of Computer Engineering

Delhi Technological University, Delhi

2013-2015

DELHI TECHNOLOGICAL UNIVERSITY

CERTIFICATE

This is to certify that the project report entitled **FUZZY CLUSTERING USING GRAVITATIONAL SEARCH ALGORITHM** is a bona fide record of work carried out by Himanshu Kaul (2K13/SWE/04) under my guidance and supervision, during the academic session 2013-2015 in partial fulfilment of the requirement for the degree of Master of Technology in Software Engineering from Delhi Technological University, Delhi.

To the best of my knowledge, the matter embodied in the thesis has not been submitted to any other University/Institute for the award of any Degree or Diploma.

Prof. OP Verma
Professor
Department of Computer Engineering
Delhi Technological University
Delhi

DELHI TECHNOLOGICAL UNIVERSITY

ACKNOWLEDGEMENTS

I feel immense pleasure to express my heartfelt gratitude to **Prof. OP Verma** for his constant and consistent inspiring guidance and utmost co-operation at every stage which culminated in successful completion of my research work.

I also would like to thank the faculty of Computer Engineering Department, DTU for their kind advice and help from time to time.

I owe my profound gratitude to my family which has been a constant source of inspiration and support.

Himanshu Kaul

Roll No. 2K13/SWE/04

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLE(S)

ABSTRACT

Clustering is a key activity in numerous data mining applications such as information retrieval, text mining, image segmentation, etc. This research work proposes a clustering approach, Fuzzy-GSA, based on gravitational search algorithm (GSA) with parameter adaptation using fuzzy inference system. The parameter α, used in the calculation of the gravitational constant G, plays a crucial role in guiding the search process in GSA. Lower values of α increase search exploration, whereas higher values increase search exploitation. In the proposed Fuzzy-GSA approach, fuzzy inference rules are used to control the value of parameter α in GSA. The performance of the Fuzzy-GSA algorithm is evaluated against four benchmark datasets. The results illustrate that the Fuzzy-GSA approach attains the highest quality clustering when compared with several other clustering algorithms.

Keywords: Clustering, Fuzzy-GSA, Gravitational Search Algorithm, Fuzzy Inference System

CHAPTER 1

INTRODUCTION

This chapter provides an introduction to clustering, fuzzy inference system and Gravitational Search Algorithm (GSA). This chapter also presents the motivation, scope and problem statement of the project. This chapter ends with a concise description of how this thesis is organised.

1.1 Basic Concepts

This section describes the fundamental concepts of clustering, fuzzy inference system and Gravitational Search Algorithm (GSA).

1.1.1 Clustering

Clustering or cluster analysis refers to the process of grouping a set of data objects such that objects belonging to the same group are similar, whereas those belonging to different groups are distinct. The final groups are called clusters or classes. It is a major data mining task and is used as a common technique for analysis of statistical data in many fields such as pattern recognition, machine learning, information retrieval etc.

In the process of clustering, it is important to define an appropriate similarity or dissimilarity measure over which the data objects are to be clustered. The same set of objects may be partitioned into different groups depending on the choice of similarity or dissimilarity measure. The number of clusters in the final partition may be pre-assigned or may be considered as an internal parameter of the clustering algorithm to be deduced based on the input data.

Clustering is an unsupervised learning task since it groups data objects into clusters without any prior information such as class labels. The clustering techniques, thus, should be able to deduce the structure embedded in data without any extra information.

Clustering algorithms have been successfully applied in several fields such as information retrieval [1,2], medicine [3], biology [4], customer analysis [5], image segmentation [6] and many others.

1.1.2 Fuzzy Inference System

Fuzzy Inference System (FIS) or fuzzy expert system is a system that maps an input space to an output space using fuzzy logic [7]. FIS employs a collection of fuzzy rules and membership functions to reason about data. The rules in FIS are of the form:

If p then q

where, p and q denote fuzzy statements and are called antecedent and consequent of the rule, respectively.

The set of rules in a Fuzzy Inference System (FIS) is known as knowledge base or rule base. Figure 1 illustrates the structure of a Fuzzy Inference System.

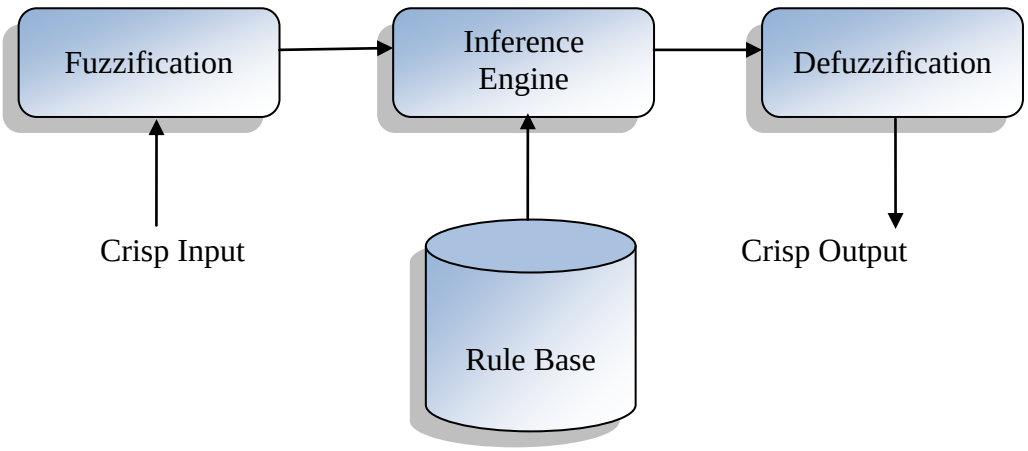

Fig 1: Fuzzy Inference System

Fuzzification converts the crisp input values to fuzzy values using membership functions defined over the input variables. Membership functions are usually designed by experts through analysis and are of several types such as triangular, trapezoidal, Gaussian, etc. Membership functions denote the degree to which a particular input value, i.e. crisp value, belongs to the corresponding fuzzy set. The degree of membership or membership value lies in the interval [0,1].

The inference engine evaluates the antecedent of each fuzzy rule and utilizes it to compute the result of each rule. In case of a single input variable, the antecedent of a fuzzy rule is same as the membership value for that input. For two or more input variables, the antecedent is computed by combining the membership values of all the input variables depending on the type of connector used in antecedent, i.e. *AND* or *OR*. If *AND* is used in the antecedent of rule, the most widely used approach is *min* method which combines different membership values in antecedent by taking the minimum of them. In case of *OR* used as a connector, the most popularly used technique is *max* method which combines the input membership values by selecting the maximum one.

After the antecedent has been computed, the inference engine calculates the result of each rule. Each rule in the rule base is assigned a weight, between 0 and 1, which denotes that rule's significance in deducing the output. Usually each rule is assigned a weight of 1. The antecedent of each rule is multiplied with the corresponding rule weight to produce that rule's result. The inference engine then aggregates the results of all the rules in order to generate a set of fuzzy outputs. The most commonly used aggregation operators are maximum, which computes point wise maximum over all of the fuzzy sets, and sum, which calculates point wise sum over all fuzzy sets.

The final step in FIS is defuzzification, which converts the fuzzy output set, obtained from inference engine, to crisp output value. The most popularly used technique for defuzzification is the centroid method. In centroid method, the crisp output value is obtained by calculating the center of gravity of the membership function for the fuzzy output set.

1.1.3 Gravitational Search Algorithm

Gravitational Search Algorithm (GSA) is an optimization algorithm proposed by Rashedi [8] in 2009. It is based on the Newton's laws of gravity and motion. The law of gravity states that "Every particle in the universe attracts every other particle with a force that is directly proportional to the product of the masses of the particles and inversely proportional to the square of the distance between them". By this definition, the gravitational force is determined using the following equation [8]:

$$F = G\,\frac{M_1 M_2}{R^2} \tag{1}$$

where, F is the gravitational force acting between two masses $M_1$and M_2, G is the gravitational constant with a value of 6.67259×10^{-11} N m^2/kg^2, and R is the distance between the two masses.

Newton's second law of motion states that when a force acts on a mass, acceleration is produced. The magnitude of acceleration produced is obtained using the equation below [8]:

$$a = \frac{F}{M} \tag{2}$$

where, F and M denote the net force acting on a given particle and its mass, respectively.

The Gravitational Search Algorithm (GSA) employs this physical phenomenon for solving optimization problems. Consider a system with N masses or agents. The position of i^{th} mass is defined as:

$$X_i = (x_i^1, \ldots, x_i^d, \ldots, x_i^n) \text{ , for } i = 1, 2, \ldots, N, \tag{3}$$

where, x_i^d is the position of i^{th} agent in d^{th} dimension and n is the total number of dimensions in the search space. The positions of agents correspond to the solutions of the

problem. The mass of each agent is computed, after evaluating the present population's fitness, using the following equations:

$$m_i(t) = \frac{fit_i(t) - \ worst(t)}{best(t) - \ worst\ (t)} \tag{4}$$

$$M_i(t) = \frac{m_i(t)}{\sum_{j=1}^{N} m_j(t)} \tag{5}$$

where, $fit_i(t)$, denotes the fitness value of i[th] agent at time *t*, and *best(t)* and *worst(t)* are computed as follows (for minimization problems):

$$best(t) = \min\ fit_j(t)\ ,\ j = 1, 2, \ldots, N \tag{6}$$

$$worst(t) = \max\ fit_j(t)\ ,\ j = 1, 2, \ldots, N \tag{7}$$

Similarly, for maximization problems *best(t)* and *worst(t)* are computed by taking the maximum and minimum fitness values respectively.

The acceleration of an agent is computed next, by considering the total forces from a set of heavier masses using the laws of gravity and motion using Equations 8 and 9. The new velocity of an agent is computed next by adding a fraction of its current velocity to its acceleration (Equation 10), followed by the calculation of its new position (Equation 11).

$$F_i^d(t) = \sum_{j \in kbest, j \neq i} rand_j\ G(t) \frac{M_j(t) M_i(t)}{R_{ij}(t) + \ \varepsilon} \left(x_j^d(t) - \ x_i^d(t)\right) \tag{8}$$

$$a_i^d(t) = \frac{F_i^d(t)}{M_i(t)} = \sum_{j \in kbest, j \neq i} rand_j\ G(t) \frac{M_j(t)}{R_{ij}(t) + \ \varepsilon} \left(x_j^d(t) - \ x_i^d(t)\right) \tag{9}$$

$$v_i^d(t+1) = \ rand_i \times v_i^d(t) + \ a_i^d(t) \tag{10}$$

$$x_i^d(t+1) = x_i^d(t) + v_i^d(t+1) \quad (11)$$

where, $rand_i$ and $rand_j$ are two random numbers uniformly distributed in the range of [0, 1], ε is a small value to prevent division by zero, $R_{ij}(t)$ is the Euclidean distance between agent *i* and agent *j*. *Kbest* is the set of first *K* agents with best fitness values and thus, largest mass. *Kbest* is dependent on time, initialized to K_o at the start and decreases as time progresses. The gravitational constant, $G(t)$, decreases with time to control the search accuracy. The value of $G(t)$ is calculated using the following equation:

$$G(t) = G_o e^{\frac{-\propto t}{T}} \quad (12)$$

where, G_o is the initial value of gravitational constant, α is a parameter which governs the degree of exploration versus exploitation of the search and *T* is the maximum number of iterations.

1.2. Motivation

Data clustering is a major task in exploratory data mining and allows for analysis of statistical data in several areas such as decision making, image segmentation, information retrieval, etc. It involves partitioning of input data into groups or clusters such that the data points within the same cluster are similar, whereas those in different clusters are dissimilar, based on certain criteria. Several clustering algorithms have been proposed in the literature to allow for an efficient partitioning.

Recent algorithms for clustering are based on evolutionary algorithms such as genetic algorithm, simulated annealing, ant colony optimization, particle swarm optimization and gravitational search algorithm [9-14]. The comparison of various evolutionary algorithms for data clustering clearly illustrates that the gravitational search algorithm outperforms the others on several real datasets [14].

However, *the Gravitational Search Algorithm (GSA) uses a constant value of parameter α for the calculation of gravitational constant.* In the beginning, smaller value of α allows for a greater exploration of the search space. Furthermore, higher value of α during the

last few iterations enhances the search space exploitation. *Therefore, the approach based on GSA can be improved by adapting and controlling the value of parameter* α *as the algorithm proceeds.*

Also, *the initial population in GSA is instantiated randomly*. Thus, the chances of reaching a global optimum are dependant over the randomly chosen initial set of agents. *An initial population which is based on the nature of a given input dataset will be of higher quality*.

From the above discussion it is evident that the clustering approach based on GSA can be refined by controlling the parameter α and selecting a higher quality initial population. This motivated us to pursue research in the field of clustering and extend the clustering algorithm based on GSA to propose an improved clustering technique.

1.3. Problem Statement

Data clustering assigns groups or clusters to input data objects, which facilitates statistical data analysis in many areas. The development of clustering algorithms has been an area of active research and several algorithms have been proposed to provide a better partitioning over input data objects. A recent research for clustering [14], based on gravitational search algorithm, provides better results when compared with many other evolutionary algorithms. However, this approach can be further refined by including provisions for:

i. Adapting the value of parameter α. In the beginning, a lower value of α is required to increase the exploration of the search space and towards the end, a higher value of α is required to enhance the exploitation of search space.

ii. Choosing a higher quality initial population, instead of it being purely random.

This research is aimed at enhancing the gravitational search algorithm based approach for clustering by incorporating the abovementioned improvements. The proposed clustering

approach uses fuzzy rules for controlling the parameter α as the search progresses. Therefore, problem of the thesis can be stated as:

Development of a technique for data clustering based on Gravitational Search Algorithm (GSA), with parameter adaptation using fuzzy inference rules.

1.4. Scope of Work

We have proposed an algorithm for clustering based on Gravitational Search Algorithm (GSA) allowing for adaptation of parameter α using fuzzy inference rules. A set of eight fuzzy rules is formulated to control the value of α on the basis of two criteria namely, current iteration number and the best fitness achieved. The current iteration numbers allows us to consider how far we have reached in the search process while controlling α. If the best fitness achieved is still high, for minimization problems, then it means we should explore further as we are still far from the desired solution. Thus, we have considered both iteration number and best fitness achieved as input variables in the fuzzy inference rules.

The proposed clustering algorithm is validated over four real datasets namely, Iris, Wine, Breast Cancer Wisconsin (Cancer) and Contraceptive Method Choice (CMC) [15]. The performance of the proposed algorithm on selected datasets is also compared with k-means [16], PSO[11], GSA [14] and combined GSA k-means approach [17]. Therefore, scope of work can be summarized as:

- Design membership functions for the input variables namely, current iteration number and the best fitness achieved, and for the output variable α.
- Formulate the fuzzy inference rules based on current iteration number and the best fitness achieved to control the parameter α.
- Validate the performance of the proposed clustering algorithm over four real datasets namely, Iris, Wine, Cancer and CMC [15].
- Compare the performance with existing clustering algorithms such as k-means, PSO, GSA and combined GSA k-means approach.

1.5. Thesis Organisation

The remaining sections of the thesis are organised as follows:

Chapter 2 presents a detailed description of different data clustering algorithms. It gives an insight to the advantages as well as disadvantages of the available clustering methods.

Chapter 3 presents the proposed algorithm for data clustering, in detail.

Chapter 4 describes the implementation aspect of this research work.

Chapter 5 demonstrates the performance of the proposed data clustering algorithm. It also compares the experimental results with other popular clustering algorithms.

Chapter 6 concludes the thesis.

CHAPTER 2

LITERATURE SURVEY

Clustering plays a very important role in data mining applications such as information retrieval, medical diagnosis, text mining and many others. It has been an area of active research and there are many algorithms proposed in the literature to perform clustering.

The most widely used and the most popular algorithm for clustering is the k-means algorithm, proposed by J. MacQueen in 1967 [16]. K-means algorithm is fairly straightforward, simple to implement and has been employed by several researchers [18-20]. However, it may be easily trapped in a local optimum and fail to achieve a global optimum in several cases since the algorithm's performance is highly dependent on the initial centroids chosen.

To overcome this problem, several heuristic based approaches have been proposed for clustering. Selim and Alsultan [10] provided a simulated annealing (SA) algorithm for clustering in 1991. They have demonstrated that the simulated annealing algorithm converges to a global optimum for the clustering problem. Maulik and Bandyopadhyay [9] presented a clustering technique based on genetic algorithm, known as GA-clustering, in 2000. The centers of a pre-defined number of clusters were encoded using chromosomes and the improved performance of GA-clustering over k-means algorithm was demonstrated with the help of three real datasets. A tabu search based method was presented for solving the clustering problem in [21,22].

Shelokar et al. presented an Ant Colony Optimization (ACO) based technique for optimally assigning objects to a pre-defined number of clusters, in 2004 [12]. The ACO based

technique provided very promising results when compared with other heuristic methods such as genetic algorithm, simulated annealing and tabu search.

Fathian et al. proposed an algorithm for clustering based on honeybee mating optimization (HBMO), in 2007 [13]. The performance of HBMO based approach was better compared to SA, GA, tabu search and ACO when evaluated over several well-known datasets. Ching-Yi et al. provided a Particle Swarm Optimization (PSO) based approach for clustering, in 2004 [11]. They compared the performance of PSO-based approach with traditional clustering algorithms and demonstrated that the PSO-based approach performed better using four artificial datasets.

Hatamlou et al. applied the Gravitational Search Algorithm (GSA) to data clustering, in 2011 [14]. The results over four well-known datasets depicted that GSA based approach performed better than several other clustering algorithms namely PSO, HBMO, ACO, GA, SA and k-means. In 2012, Hatamlou et al., presented a technique combining the benefits of k-means algorithm with GSA, called GSA-KM, in clustering [17]. In GSA-KM approach, the initial population for GSA was generated with the help of k-means algorithm which allowed GSA to converge faster. When compared with other well known algorithms, such as k-means, GA, SA, ACO, HBMO, PSO and the conventional GSA approach, GSA-KM approach provided better results over several real datasets.

CHAPTER 3

PROPOSED APPROACH FOR CLUSTERING

In this chapter, we describe the proposed method, called *Fuzzy-GSA*, for data clustering. The proposed approach is based on Gravitational Search Algorithm (GSA), described in Chapter 1, and uses fuzzy inference rules for controlling the parameter α as search progresses. This chapter is divided into two sections. The first section describes the Fuzzy Inference System (FIS) developed, and the second section presents the proposed Fuzzy-GSA algorithm for clustering.

3.1 The Developed Fuzzy Inference System

The FIS is developed with two input variables and one output variable. The input variables are as follows:

- *IT*: The current iteration number.
- *Fbest*: The best value of fitness achieved till the current iteration.

IT enables us to consider how far we have reached in the search process. During the initial iterations, i.e. when *IT* is low, a lower value of α is desired since lower the value of α, higher the value of gravitational constant, *G(t)*, will be (Equation 12) and thus, higher the force, *F*, (Equation 8) resulting in a higher acceleration, *a*, (Equation 9) and velocity, *v(t)* (Equation 10). This allows for higher exploration at the beginning of search. Similarly, towards the final few iterations, i.e. when *IT* is high, a higher value of α is desired to promote higher exploitation. Figure 2 depicts the membership function for *IT*. The iterations are represented as a fraction of the maximum number of iterations allowed, such that 0.5 means half of the total iterations and 1 represents the maximum iterations.

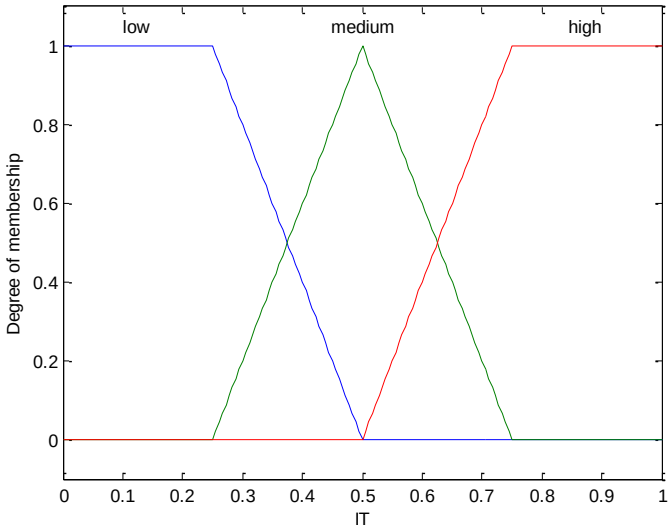

Fig 2: Membership Function for *IT*

Fbest represents the lowest value of fitness, since clustering is a minimization problem with the fitness function as mean square error, achieved till the current iteration. If the value of *Fbest* is high, then we need to reduce α to promote a greater exploration, since higher values for *Fbest* mean we are still far from the solution. However, if *Fbest* is low, we should increase α to allow for a higher exploitation as we are near the solution.

Note that the fitness function, representing the total mean square error or the sum of intra-cluster distances, is computed using the following equation [23]:

$$f(O,C) = \sum_{l=1}^{k} \sum_{O_i \in C_l} d(O_i, CC_l)^2 \tag{13}$$

where, CC_l represents the centroids of the cluster C_l , $d(O_i, CC_l)$ denotes the distance or dissimilarity between object O_i and cluster centroid CC_l. The most popular and widely used distance metric is the Euclidean distance, which we have used in this work. Euclidean distance between two objects X_i and X_j with d dimensions is calculated as:

$$d(X_i, X_j) = \sqrt{\sum_{p=1}^{d} (x_i^p - x_j^p)^2} \tag{14}$$

where, x_i^p denotes the value of p^{th} dimension for the object X_i and x_j^p denotes the value of p^{th} dimension for the object X_j.

The developed FIS consists of one output variable, i.e. *alpha(t)*, which denotes the value of parameter α in Equation 12. Figure 3 shows the membership function for *alpha(t)*. The range of parameter α is taken as [0, 50] to provide a wide range of search on the value of *alpha(t)*.

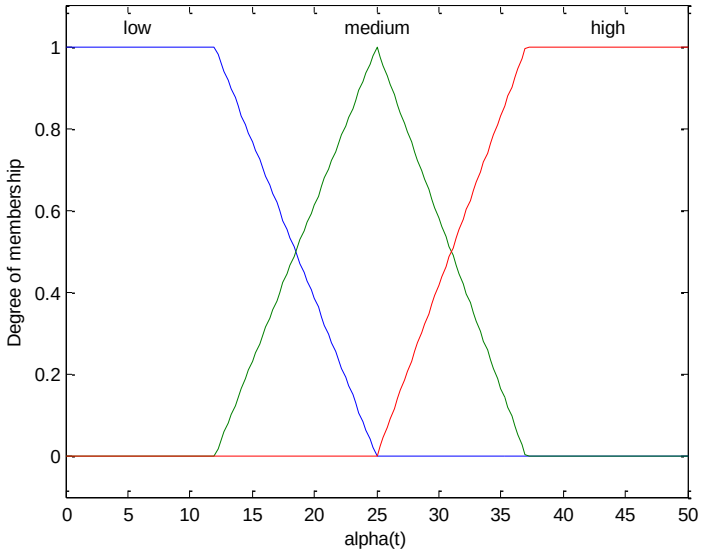

Fig 3: Membership Function for *alpha(t)*

The following eight fuzzy rules were formulated to control the parameter α in the calculation of the gravitational constant (Equation 12):

i. If (*IT* is low) and (*Fbest* is low) then (*alpha(t)* is high)
ii. If (*IT* is low) and (*Fbest* is medium) then (*alpha(t)* is medium)
iii. If (*IT* is low) and (*Fbest* is high) then (*alpha(t)* is low)
iv. If (*IT* is medium) and (*Fbest* is high) then (*alpha(t)* is low)
v. If (*IT* is medium) and (*Fbest* is medium) then (*alpha(t)* is medium)
vi. If (*IT* is high) and (*Fbest* is high) then (*alpha(t)* is medium)
vii. If (*IT* is high) and (*Fbest* is medium) then (*alpha(t)* is medium)
viii. If (*IT* is high) and (*Fbest* is low) then (*alpha(t)* is high)

Figure 4 depicts the developed fuzzy inference system with two inputs, one output and eight inference rules. The method used for "And" was min and for "Or" was max. The implication method was min, aggregation method was max and defuzzification method was centroid.

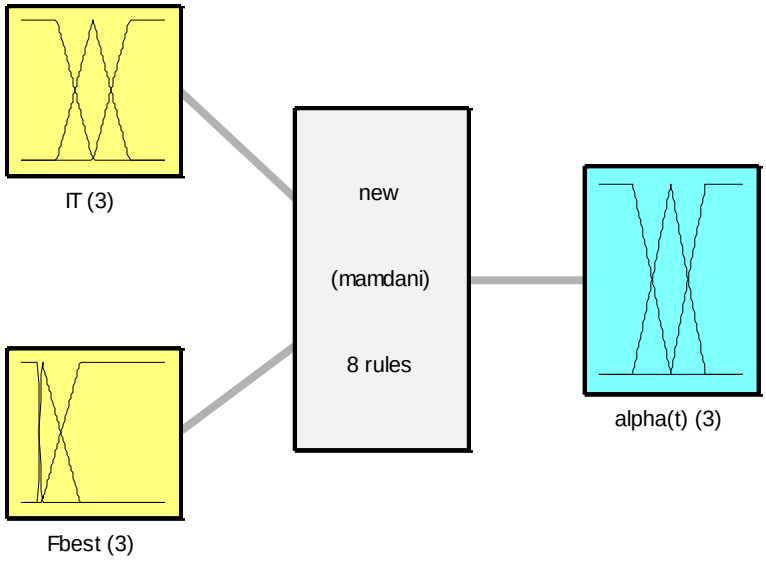

Fig 4: The Developed Fuzzy Inference System

3.2 Proposed Algorithm

The proposed algorithm, *Fuzzy-GSA*, comprises of two main steps. The first step is to generate an initial population for GSA. The proposed method provides a better initial population which would allow for a higher exploration since a wide range of values is present while searching the solution space. The rest of the agents are generated randomly by considering the range of features in the given dataset.

The second step involves application of GSA, described in Chapter 1, to the given dataset and using the fuzzy inference system developed to control the parameter α while searching for the solution.

The step by step algorithm for the proposed approach is stated next. Let N denote the population size, C_i be the i^{th} candidate solution or agent, k be the number of clusters, d be the number of features in a given dataset.

Step 1: Generate initial population, $P = \{C_1, C_2, \ldots, C_N\}$.

- Generate C_1 consisting of maximum values of all the features.
- Generate C_3 consisting of median values of all the features.
- Generate the remaining N-3 candidates randomly within the range of minimum to maximum values for all features.

Step 2: Apply GSA and use the developed FIS, described in Section 3.1, for parameter adaptation.

- Calculate the fitness function, as per Equation 13, for all the candidate solutions.
- Feed the values of *IT*, current iteration number, and *Fbest*, best fitness achieved, as inputs to the developed FIS, and obtain the value of parameter α.
- Calculate G, F, M and a for all the candidate solutions using Equations 5, 8, 9 and 12 as described in the Gravitational Search Algorithm (GSA).
- Update the velocity and position of each candidate solution as per Equation 10 and 11 respectively.
- Check if termination criteria, i.e. maximum number of iterations allowed is reached or fitness function is not exhibiting a minimum improvement, are met. If yes, then return the best value of fitness function achieved as the final solution, else reiterate through step 2.

The final solution consists of the best value of fitness function, i.e. the minimum mean square error, achieved by running the proposed Fuzzy-GSA algorithm.

CHAPTER 4

IMPLEMENTATION

This chapter provides the implementation details of the proposed Fuzzy-GSA clustering algorithm. The detailed explanation pertaining to implementation can be divided into two sections. The first section provides a brief description of the implementation platform and the second section discusses the operational details of the proposed algorithm.

4.1 Brief Description

The proposed Fuzzy-GSA algorithm for clustering is implemented in MATLAB 7.7.0.471 on an i5-3337U computer at 1.80 GHz processor with 4 GB RAM. Figure 5 shows the results obtained by running the proposed algorithm over Iris dataset [15]. As can be seen the Fuzzy-GSA algorithm converges at 177 iterations and achieves the minimum fitness value of 96.5403.

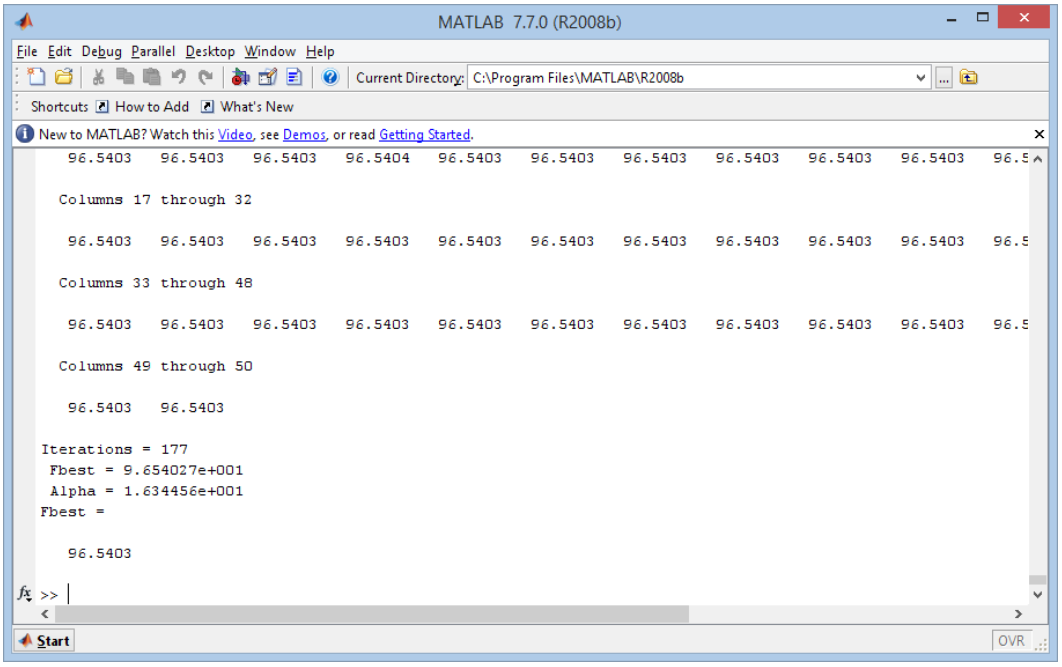

Fig 5: Execution over Iris dataset

Figure 6 demonstrates the changes in the value of best fitness achieved so far over Iris dataset, as the Fuzzy-GSA algorithm proceeds. The X- axis of the graph represents the iteration number and the Y-axis represents the best fitness achieved. As can be seen, the best value of fitness function so far drops and converges to a minimum value before 180 iterations.

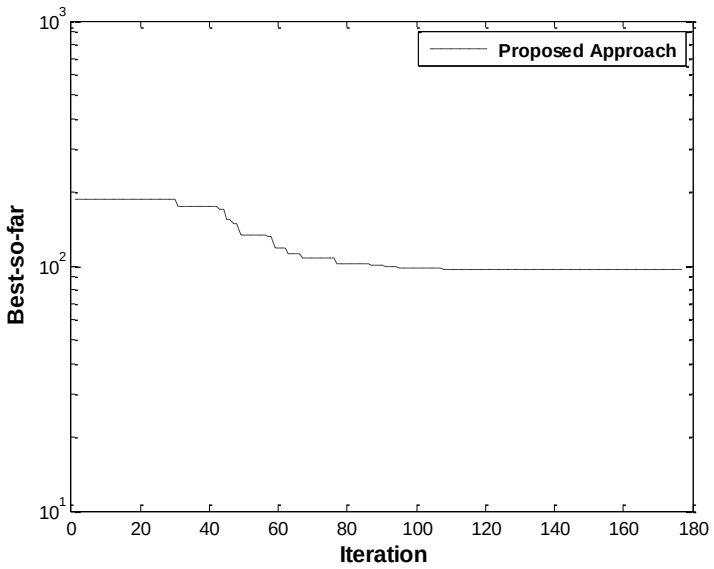

Fig 6: Best-so-far versus Iteration Graph

4.2 Implementation Details of Proposed Algorithm

Each candidate solution, in the population, consists of cluster centers for each of the k clusters, and each cluster center comprises of values for each feature in a dataset. Figure 7 illustrates the representation of the i^{th} candidate solution C_i. CC_{ij} denotes the j^{th} cluster center of the i^{th} candidate solution and F_{ij} represents the value of j^{th} feature for i^{th} cluster center. Therefore, each candidate solution consists of $(d \times k)$ values.

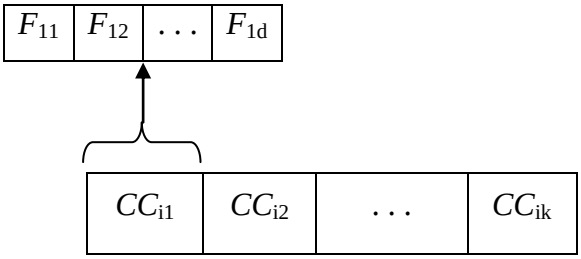

Fig 7: Representation of i^{th} Candidate Solution, C_i

The value of parameter α, represented by variable *alpha(t)* in the developed FIS, obtained for given values of *IT* and *Fbest* is depicted in Figure 8.

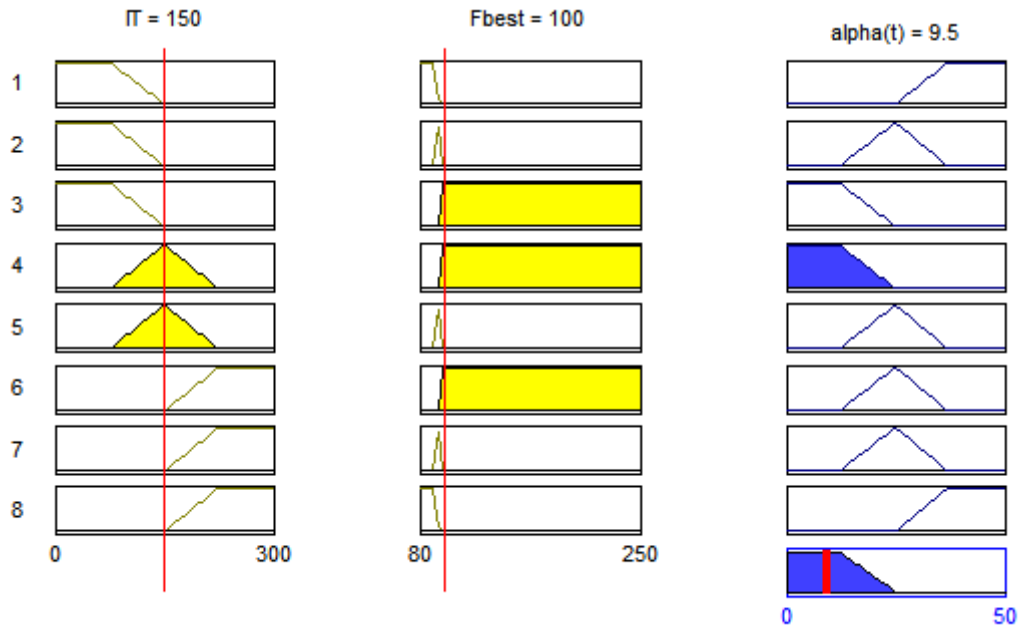

Fig. 8: Calculation of α by the developed FIS

CHAPTER 5

EVALUATION AND RESULTS

This chapter discusses evaluation of the proposed Fuzzy-GSA algorithm for clustering. This chapter is divided into two sections. The first section describes the performance evaluation of the proposed approach and the second section presents the results of evaluation.

5.1 Performance Evaluation of the Proposed Algorithm

We have measured the performance of the proposed approach, Fuzzy-GSA, by calculating the sum of intra-cluster distances or mean square error (MSE), as defined by Equation 13. We have considered four benchmark datasets namely, Iris, Wine, Breast Cancer Wisconsin and Contraceptive Method Choice (CMC) for evaluation. The datasets are all obtained from UC Irvine repository of machine learning databases [15] and have been extensively used by researchers to validate the performance of clustering algorithms. A description of each benchmark dataset is provided below:

- *Iris Dataset*: It consists of three classes with 50 instances each, where each class refers to a species of iris flower. There are four features in the dataset namely, petal length, petal width, sepal length and sepal width which report certain characteristics of iris flower. The dataset comprises of a total of 150 instances. There no missing feature values in this dataset.

- *Wine Dataset*: It consists of three classes representing different types of wine. The data is a result of a chemical analysis of wines grown in the same region in Italy but derived from three different cultivators. There are 13 features which represent quantities of different constituents found in each of the three types of wines. The dataset consists of 178 instances with no missing values.

- *Breast Cancer Wisconsin Dataset*: This dataset comprises of two classes namely, malignant and benign representing the severity of cancer. There are a total of 683 instances, without missing values. It has 9 attributes or features namely, clump thickness, uniformity of cell size, uniformity of cell shape, marginal adhesion, single epithelial cell size, bare nuclei, bland chromatin, normal nucleoli and mitoses.

- *Contraceptive Method Choice (CMC) Dataset*: It consists of three classes namely, no-use, long-term and short-term. There are 1473 instances in this dataset, without any missing values. It contains 9 features or attributes namely, wife's age, wife's education, husband's education, number of children ever born, wife's religion, whether wife's working, husband's occupation, standard of living and media exposure.

Sum of intra-cluster distances is then calculated over each of the four benchmark datasets considered, using Equation 13. We have also compared the performance of the proposed Fuzzy-GSA approach with existing clustering algorithms such as conventional GSA [14], combined k-means and GSA [17], PSO [11] and k-means [16] algorithms on selected datasets. Due to the stochastic nature of these algorithms, we have considered 20 independent runs for each algorithm over each dataset.

The results are then compared in terms of best, average and worst solutions over 20 independent simulations. Moreover, the standard deviation of the achieved solutions from each clustering algorithm is also calculated. Note that, a lower value of the sum of intra-cluster distances denotes a higher quality clustering.

5.2 Results

This section presents the results of evaluation of the proposed clustering approach, i.e. Fuzzy-GSA, over the four benchmark datasets described in Section 5.1. The performance of the proposed Fuzzy-GSA algorithm is also compared with existing clustering algorithms such as conventional GSA [14], combined k-means and GSA [17], PSO [11] and k-means [16] algorithms over the selected datasets.

Figure 9 shows the graphical representations of the sum of intra-cluster distances obtained by the proposed Fuzzy-GSA algorithm over 20 independent runs. The X- axis of the graphs represents the independent simulations and the Y-axis represents sum of intra-cluster distances. The colours blue, red, green and purple represent Iris, Wine, Breast Cancer Wisconsin and CMC datasets, respectively. It can be observed from Figure 9(a), 9(b), 9(c) and 9(d) that the proposed approach returns nearly constant values as solutions without much deviation in results for each of the four benchmark datasets.

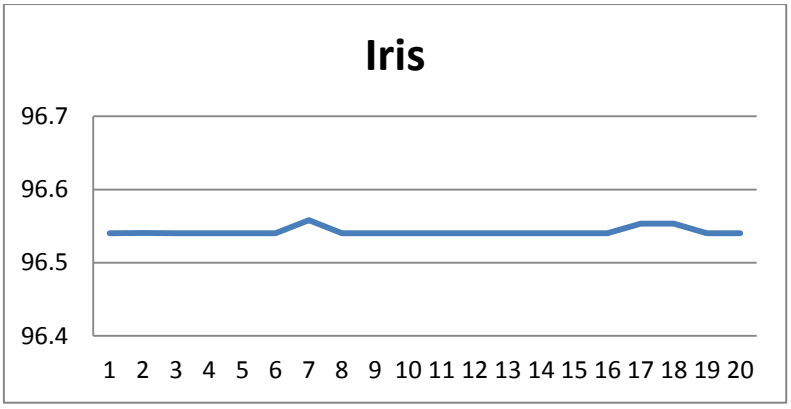

Fig. 9(a): Performance of Fuzzy-GSA over Iris Dataset

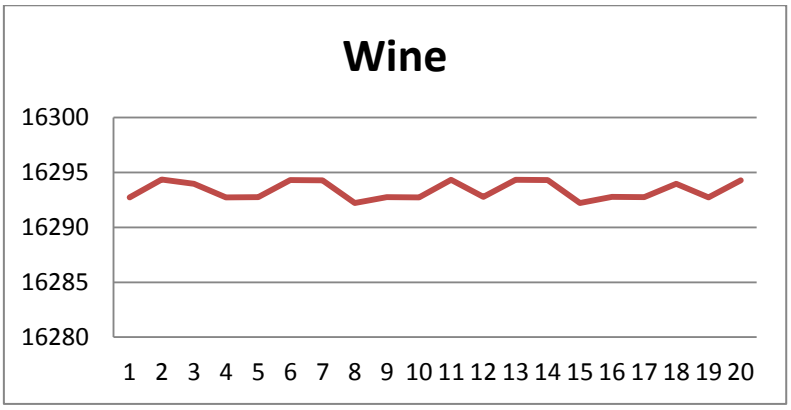

Fig. 9(b): Performance of Fuzzy-GSA over Wine Dataset

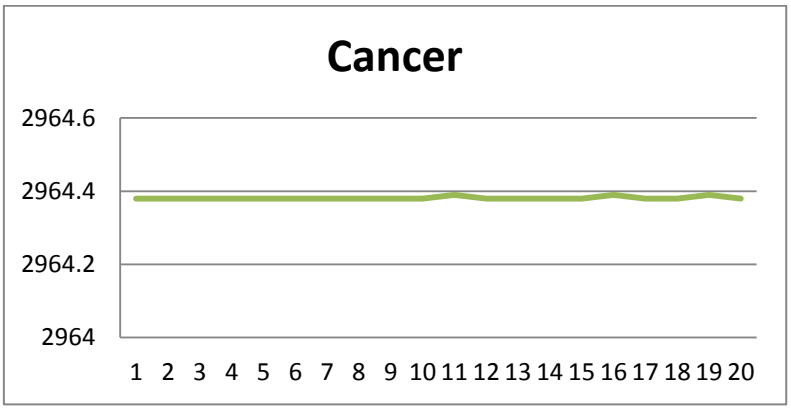

Fig. 9(c): Performance of Fuzzy-GSA over Cancer Dataset

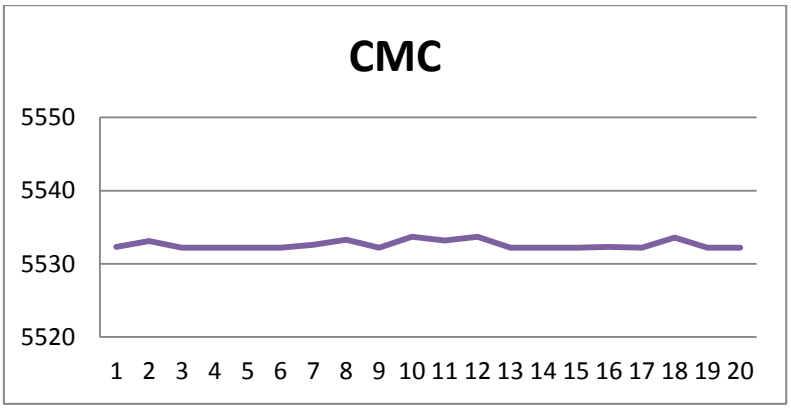

Fig. 9(d): Performance of Fuzzy-GSA over CMC Dataset

Figure 12 illustrates the three clusters assigned by the proposed Fuzzy-GSA algorithm over the Iris dataset. It shows a 3-D plot considering the three dimensions namely, petal length, sepal width and sepal length of the Iris dataset. The remaining dimension, petal width, is highly correlated with the dimension, petal length, and thus can be ignored without losing the quality of cluster representation. The three clusters, corresponding to the three types of iris flower, are depicted by three different colour coded symbols namely, blue squares, green circles and red triangles. The X-axis represents the dimension sepal length, the Y-axis

represents the dimension sepal width and the Z-axis represents the dimension petal length. The respective cluster centers are represented by black coloured circles.

Fig 10: Cluster plot by Fuzzy-GSA over Iris Dataset

The best, average, worst and standard deviation of the obtained solutions by different clustering algorithms on the selected datasets are shown in Table 1. The results are achieved over 20 independent runs.

Table 1: Simulation Results for Clustering Algorithms

Dataset	**Criteria**	**K-means**	**PSO**	**GSA**	**GSA-KM**	**Fuzzy-GSA**
Iris	Best	97.2046	96.7170	96.6700	96.6173	96.5403
	Avg	101.2562	97.8962	96.6952	96.6687	96.5425
	Worst	124.2155	99.7773	96.8961	96.6989	96.5581
	Std	9.8954	0.9306	0.0485	0.0227	0.0054

Wine	Best	16555.6794	16340.1288	16319.0752	16300.0862	16292.23
	Avg	16990.4711	16378.4879	16351.3308	16301.6686	16293.369
	Worst	18294.8465	16505.4147	16481.6366	16302.5723	16294.35
	Std	772.6452	44.7783	34.3939	0.6280	0.8200
Cancer	Best	2988.4278	2974.6453	2965.1822	2965.0778	2964.38
	Avg	2988.4278	3078.4729	2975.0247	2965.6777	2964.38
	Worst	2988.4278	3336.6453	2997.7815	2966.7573	2964.39
	Std	0	113.8010	8.8999	0.4191	0.003
CMC	Best	5543.5119	5710.8682	5544.6439	5543.5119	5532.2
	Avg	5543.7652	5840.8038	5627.3252	5544.5250	5532.6
	Worst	5545.2005	5987.0105	5697.1460	5545.2005	5533.7
	Std	0.6186	82.3954	48.8495	0.8487	0.5831

As can be seen from Table 1, the proposed approach, Fuzzy-GSA, demonstrates the highest quality solutions in terms of best, average and worst intra-cluster distances over all the four benchmark datasets. Furthermore, the standard deviation of Fuzzy-GSA is smaller, which indicates that it can locate a near-optimal solution in most of the cases when compared with other clustering algorithms.

For Iris dataset, the best, average and worst solutions by the proposed Fuzzy-GSA approach are 96.5403, 96.5425 and 96.5581, respectively with a standard deviation of 0.0054. For Wine dataset, the best, average and worst solutions achieved by the Fuzzy-GSA approach are 16292.23, 16293.369 and 16294.35 respectively, with a standard deviation of 0.82. For Breast Cancer Wisconsin dataset, the achieved best, average and worst solutions are 2964.38, 2964.38 and 2964.39, respectively with a standard deviation of 0.003. Lastly, for CMC dataset, the best, average and worst solutions obtained by Fuzzy-GSA are 5532.2, 5532.6 and 5533.7, respectively with a standard deviation of 0.5831.

To summarize, the proposed Fuzzy-GSA approach achieves the best quality clustering when compared with several popular clustering algorithms, depicted in Table 1, over four benchmark datasets considering 20 independent runs.

CHAPTER 6

CONCLUSIONS

This research work proposes an algorithm, *Fuzzy-GSA*, for clustering. *Fuzzy-GSA* algorithm is based on the conventional Gravitational Search Algorithm (GSA) with a provision for adapting the value of parameter α used in the calculation of the gravitational constant. In the beginning, a smaller value of α is desired to achieve a higher exploration, whereas towards the end of search, a relatively higher value of α helps in achieving a higher exploitation. *Fuzzy-GSA* algorithm incorporates Fuzzy Inference System (FIS) into the conventional GSA to allow for parameter adaptation. The parameter α is controlled by using eight formulated fuzzy inference rules, in *Fuzzy-GSA*.

Also, *Fuzzy-GSA* algorithm generates a better quality initial population for GSA by considering the nature of dataset being considered. It generates three candidate solutions consisting of maximum, minimum and median values, respectively in a dataset thereby building an initial population which covers a wider range. This helps in achieving a higher exploration.

The performance of *Fuzzy-GSA* is evaluated by comparing its best, average and worst solutions with several other clustering algorithms over four selected benchmark datasets namely, Iris, Wine, Breast Cancer Wisconsin and CMC, considering 20 independent runs. The results show that *Fuzzy-GSA* achieves the highest quality clustering with very small standard deviation, when compared with several other clustering algorithms.

REFERENCES

[1] N. Jardine, C.J. van Rijsbergen, “The use of hierarchic clustering in information retrieval”, Information Storage and Retrieval, vol. 7, issue 5, pp. 217–240, 1971.

[2] A. Tombros, R. Villa, C.J. van Rijsbergen, “The effectiveness of query-specific hierarchic clustering in information retrieval”, Information Processing & Management, vol. 38, issue 4, pp. 559–582, 2002.

[3] L. Liao, T. Lin, B. Li, “MRI brain image segmentation and bias field correction based on fast spatially constrained kernel clustering approach”, Pattern Recognition Letters, vol. 29, issue 10, pp. 1580–1588, 2008.

[4] G. Kerr, H.J. Ruskin, M. Crane, P. Doolan, “Techniques for clustering gene expression data”, Computers in Biology and Medicine, vol. 38, issue 3, pp. 283–293, 2008.

[5] B. Saglam, M. Türkay, F.S. Salman, S. Sayın, F. Karaesmen, E.L. Örmeci, “A mixed-integer programming approach to the clustering problem with an application in customer segmentation”, European Journal of Operational Research, vol. 173, issue 3, pp. 866–879, 2006.

[6] Y. Xia, D. Fenga, T. Wangd, R. Zhaob, Y. Zhangb “Image segmentation by clustering of spatial patterns”, Pattern Recognition Letters, vol. 28, issue 12, pp. 1548–1555, 2007.

[7] John Yen and Reza Langari, Fuzzy Logic – Intelligence, Control and Information, Prentice Hall, 1st edition.

[8] E. Rashedi, H. Nezamabadi-pour, S. Saryazdi, “GSA: a gravitational search algorithm”, Information Sciences, vol. 179, issue 13, pp. 2232–2248, 2009.

[9] U. Maulik, S. Bandyopadhyay, “Genetic algorithm-based clustering technique”, Pattern Recognition, vol. 33, issue 9, pp. 1455–1465, 2000.

[10] S.Z. Selim, K. Alsultan, “A simulated annealing algorithm for the clustering problem”, Pattern Recognition, vol. 24, issue 10, pp. 1003–1008, 1991.

[11] C. Ching-Yi, Y. Fun, “Particle swarm optimization algorithm and its application to clustering analysis”, in IEEE International Conference on Networking, Sensing and Control, 2004.

[12] P.S. Shelokar, V.K. Jayaraman, B.D. Kulkarni, “An ant colony approach for clustering”, Analytica Chimica Acta, vol. 509, issue 2, pp. 187–195, 2004.

[13] M. Fathian, B. Amiri, A. Maroosi, “Application of honey-bee mating optimization algorithm on clustering”, Applied Mathematics and Computation, vol. 190, issue 2, pp. 1502–1513, 2007.

[14] A. Hatamlou, S. Abdullah, H. Nezamabadi-pour, “Application of gravitational search algorithm on data clustering”, Rough Sets and Knowledge Technology, Springer, pp. 337–346, 2011.

[15] C.L. Blake, C.J. Merz, UCI repository of machine learning databases. Available from: http://www.ics.uci.edu/-mlearn/MLRepository.html.

[16] J. MacQueen, “Some methods for classification and analysis of multivariate observations”, In Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability, vol. 1, pp. 281-297, 1967.

[17] A. Hatamlou, S. Abdullah, H. Nezamabadi-pour, “A combined approach for clustering based on K-means and gravitational search algorithms”, Swarm and Evolutionary Computation, vol. 6, pp. 47–52, 2012.

[18] A.K. Jain, “Data clustering: 50 years beyond K-means”, Pattern Recognition Letters, vol. 31, issue 8, pp. 651–666, 2010.

[19] E.W. Forgy, “Cluster analysis of multivariate data: efficiency versus interpret- ability of classifications”, Biometrics, vol. 21, issue 2, 1965.

[20] L. Kaufman, P.J. Rousseeuw, “Finding Groups in Data: An Introduction to Cluster Analysis”, John Wiley & Sons, New York, 1990.

[21] K.S. Al-Sultan, “A tabu search approach to the clustering problem”, Pattern Recognition, vol. 28, issue 9, pp. 1443–1451, 1995.

[22] C.S. Sung, H.W. Jin, “A tabu-search-based heuristic for clustering”, Pattern Recognition, vol. 33, issue 5, pp. 849–858, 2000.

[23] S. Yang, R. Wu, M. Wang, L. Jiao, “Evolutionary clustering based vector quantization and SPIHT coding for image compression”, Pattern Recognition Letters, vol. 31, issue 13, pp. 1773–1780, 2010.