World of Computer Science and Information Technology Journal (WCSIT)
ISSN: 2221-0741
Vol. 2, No. 4, 131-136, 2012

Hybridization Between Iterative Simulated Annealing and Modified Great Deluge for Medical Clustering Problems

Anmar Abuhamdah
Department of Computer Science
Jearsh private university
Amman, Jordan

Bassam M. El-Zaghmouri
Department of Computer Information Systems
Jearsh private university
Amman, Jordan

Anas Quteishat
Department of Computer Engineering,
Al- Balqa' Applied University
Amman, Jordan

Rawnaq Kittaneh
Department of Computer Science
University Kabangsaan Malaysia
Bangi, Malaysia

Abstract—Clustering is a type of classification under optimization problems, which is considered as a critical area of data mining. Medical clustering problem is a type of unsupervised learning in data mining. This work present a hybridization between our previous proposed Iterative Simulated Annealing (ISA) and Modified Great Deluge (MGD) algorithms for medical clustering problems. The aim of this work is to produce an effective algorithm for partitioning N objects into K clusters. The idea of the hybridization between MGD and ISA is to incorporate the strength of one approach with the strength of the other hoping a more promising algorithm. Also this combination can help to diverse the search space. Experimental results obtained two way of calculating the minimal distance that have been tested on six benchmark medical datasets show that, ISA-MGD is able to outperform some instances of MGD and ISA algorithms.

Keywords- Clustering; Modified Great Deluge component; Iterative Simulated Annealing.

I. INTRODUCTION

Clustering problem is considered as an important type of unsupervised data mining technique, which divides the input space into *K* regions based on some similarity / dissimilarity measures, where the *K* value might not be known priori [1], [2]. A cluster is a set of data (objects) which are related to each other within the same cluster and unrelated to the data in other clusters [1], [2]. The process of grouping a set of physical data into classes of related data is called clustering [1], [2]. Clustering problem is an NP-hard problem when there are more than three clusters [3]. Recently, the problem with two clusters also considered as an NP-hard problem [4], [5].

Clustering in a problem aims to minimize within the group variance and maximize between the group variance [6]. Recently, there are various algorithms has been applied on numerous domains to solve clustering problems, like Genetic Algorithm [7], [8] and [9], Tabu Search [10], and Artificial Bee Colony [11], iterative simulated annealing [12] and modified great deluge (MGD) [13].

In this work, we use a *Multi K-Means* algorithm as in [12] to construct the initial solution since it simple and able to deal with a huge number of data patterns.

In this work, we hybridize Modified Great Deluge (MGD) [13] and Iterative Simulated Annealing (ISA) [12] algorithms to diverse the search space of MGD and ISA algorithms and incorporate the strength of each other. The idea of the hybridization (ISA-MGD) is to overcome some of the limitation of MGD and ISA algorithms.

In order to evaluate the performance of the propose algorithm, ISA-MGD applied over six benchmark datasets using two ways of calculation the minimal distance (i.e. between objects and between centers). After that, it's compared to MGD and extended ISA algorithms that applied by using the same two ways of calculation the minimal distance and the six benchmark datasets [12], [13].

II. PROBLEM DESCRIPTION

This work spotlight on clustering problems using medical datasets where there are six well-known public domain

benchmark datasets (see section A) that are presented in UCI machine learning repository (http://archive.ics.uci.edu/ml/index.html) are used to evaluate the performance of the proposed approaches in this work by using two way of calculation the minimal distance (see section B). For further information on medical clustering problems, please refer to [12], [2].

A. Benchmark Datasets

In general, there are varied datasets in terms of the records number and attributes that have different complexity. All of the datasets information about the diseases that are taken from a real infected patients and denoted for research purposes. . The number of clusters in all of the datasets tested here is recommended by the provider of the datasets, except one dataset was recommended [2]. Datasets are summarized as follow:

Dataset 1: Wisconsin Breast Cancer Database (*B.C*) with:

- number of instances: 699
- number of attributes: 10 (including the class attribute)
- attributes type: integers
- number of clusters used in *K-Means* is 3 (i.e. 123, 240 and 363).

Dataset 2: Lung Cancer Database (*L.C*) with:

- number of instances: 32
- number of attributes: 56
- attributes type: integers
- number of clusters used in *K-Means*: 3 (i.e. 9, 13 and 10).

Dataset 3: BUPA Liver Disorders Database (*B.L.D*) with:

- number of instances: 345
- number of attributes: 7
- attributes type: integer, categorical and real
- number of clusters used in *K-Means*: 2 (i.e. 145 and 200).

Dataset 4: Pima Indian Diabetes Database (*P.I.D*) with:

- number of instances: 768
- number of attributes: 8
- attributes type: integer and real
- number of clusters used in *K-Means*: 2 (i.e. 500 and 268).

Dataset 5: Haberman's Survival Database (*H.S*) with:

- number of instances: 306
- number of attributes: 4
- attributes type: integer
- number of clusters used in *K-Means*: 2 (i.e. 225 and 81).

Dataset 6: Thyroid gland data Disease Database (*T.D*) with:

- number of instances: 215
- number of attributes: 21
- attributes type: categorical and real
- number of clusters used in *K-Means*: 3 (i.e. 150, 30 and 35).

B. Cluster Quality Calculation

In cluster analysis, it is very important issue to evaluate the clustering results quality that is produced by a certain measure. Those measures can be used to compare solutions obtained from different algorithms and also can be used to guide some optimization search processes to find the best partitioning procedure which fits the underlying dataset [14], [15] and [16].

In this work, two functions are used to measure the clusters and their centers i.e.:

i. Between Objects
ii. Between Centers

These measurement on the cluster objects and / or changes on the cluster centers have the direct effect to the cluster quality. The two types of functions are portrayed as below.

i. Between Objects

This method depends on the calculation on the distance (cluster quality) between each data pattern. So we produced a method for clustering in which samples are added in if they are ''close'' to at least one sample in the candidate cluster.

ii. Between Centers

This method depends on calculating the cluster quality (distance) using the sum of distance between each of data pattern and the cluster center that it belongs to.

III. THE ALGORITHM

In this work, we use two (i.e. N1 and N2) neighborhood structures (see section A) to propose the hybridization between Iterative Simulated Annealing (ISA) and Modified Great Deluge Algorithm (MGD) algorithm is presented for solving clustering problems (ISA-MGD). (see section B).

A. Neighborhood Structures

In this work, two neighborhood structures are used which are adopted from [17] (coded as N1 and N2). The description of the employed neighborhood structures are given as follows:

N1: Randomly selects one pattern from each cluster to swap their data as in [12].

N2: Randomly select two different patterns from the same cluster and swap their data as in [12].

B. Hybridization between iterative simulated annealing and modified great deluge algorithms

In this work, the hybridization between Iterative Simulated Annealing (ISA) and Modified Great Deluge Algorithm (MGD) algorithm is presented for solving clustering problems (ISA-MGD). This hybridization aims to incorporate the strength of one approach with the strength of the other hoping a more promising algorithm. Also this combination can help to diverse the search space.

The Iterative Simulated Annealing (ISA) is proposed in

[12] to overcome the limitation of the original Simulated Annealing (SA) [1]. ISA structure resembles SA structure, but the basic difference is in term of the stopping condition. In ISA, when the temperature (*Temp*) is less or equal to the final temperature, we reinitialize the temperature (*Temp*) equal to the initial temperature (T_0). This process of reinitializing the temperature will be repeated in ISA, until no improvement obtained on the best solution ($S_{Arranget}$). Where, Modified Great Deluge (MGD) is proposed in [13] to overcome the limitation of the original GD [13]. MGD the original structure of the GD algorithm, but the basic difference is in term of updating the *level*. In MGD, we have introduce a list that keeps the previous *level* value at the time when the better solution is obtained (i.e. $S_{Arrange} = S_{working}$). When the maximum number of iteration of no improved GD (*not_improving_length_GD*) is met, then the *level* is updated by a new *level* that is randomly selected from the list (where the size of the list is set to 10 based on preliminary experiments)

In this work, a hybridization between iterative simulated annealing and the modified great deluge algorithms are applied, where the algorithm starts with a given *K-Means* partitions. The notations used in this work are listed as below:

- S_o : initial solution
- $f(S_o)$: quality of S_o
- T_o : initial temperature
- T_f : final temperature
- *Temp* : current temperature, where at first is set to T_o
- α : decreasing rate
- $S_{Arrange}$: best solution
- $f(S_{Arrange})$: the quality of $S_{Arrange}$
- S_{source} : the current solution
- $f(S_{source})$: the quality of S_{source}
- $S_{working}$: the candidate solution
- $f(S_{working})$: the quality of $S_{working}$
- *level* : boundary
- *est.q* : estimated quality of the final solution
- *N.iters* : number of iterations
- *Iterations* : iteration counter
- β : decreasing value
- *LL* : list to store level values
- *not_improving_length_GD*: maximum number of iterations where there is not improvement in the quality of the solution

The same parameters as those employed in ISA [12] and MGD [13] are imposed in this work, where the initial temperature T_o is equal to 10, and the final temperature T_f is 0. At the beginning of the search, *Temp* is set to be T_o, and at every iteration the temperature *Temp* is decreased by α, where α is equal to 0.7. The *level* is set to be an initial water level. The *level* is decreased by β in each of the iterations, where β is based on the estimated quality (*est.q*). The pseudo code for the ISA-MGD for clustering problems is shown in Fig. 1.

From Fig. 1, the algorithm starts by initializing the required parameters as in Step-1 by setting the stopping condition (*N.iters*), the temperature (*Temp*) is equal to the initial temperature (T_o), initialize the decreasing temperature rate (α), initialize the estimated quality of the final solution (*est.q*), the initial water level (*level*), decreasing rate (β), maximum number of not improving solutions (*not_improving_length_GD)* and a list of *LL* size to store the value of the *level*. Note that, the initial solution is generated using *K-Means* (S_o).

Procedure ISA-MGD Algorithm
Step 1: Initialization Phase
Determine initial candidate solution S_o and $f(S_o)$;
$S_{Arrange} = S_o$; $f(S_{Arrange}) = f(S_o)$; $S_{source} = S_o$; $f(S_{source}) = f(S_o)$;
Set N.iters; (stopping condition)
Set estimated quality of final solution, est.q;
Set not_improving_length_GD;//maximum number of GD not improved
level= $f(S_o)$; decreasing rate $\beta = ((f(S_o) - est.q) / (N.iters))$;
Set initial temperature T_o; Set final temperature T_f; Set Temp = T_o;
Set decreasing temperature rate as α, where α = 0.7;
Set list size, to store level values into list LL;
Iterations=0; not_improving_counter=0;
Step 2: Improvement (Iterative) Phase
repeat *(while termination condition is not satisfied)*
Step 2.1: *Selecting candidate solution $S_{working}$*
Generate candidate solutions by applying neighbourhood structure (N1 and N2) and the best solution consider as candidate solution ($S_{working}$);
Step 2.2: Accepting Solution
if *$f(S_{working}) < f(S_{Arrange})$*
$S_{Arrange} = S_{working}$; $f(S_{Arrange}) = f(S_{working})$;
$S_{source} = S_{working}$; $f(S_{source}) = f(S_{working})$;
Update LL by the level value (FIFO);
not_improving_counter = 0;
else
$\delta = f(S_{working})$ - quality (S_{source})
Generate a random number called RN between 0 and 1;
if *$RN \leq e^{-\delta/Temp}$*
$S_{source} = S_{working}$;
else
if *$f(S_{working}) \leq level$*
$S_{source} = S_{working}$;
else
Increase not_improving_counter by one;
if *not_improving_counter ==not_improving_length_GD,*
Temp = T_o; Level = random level from the list LL;
end if
*Temp = Temp-Temp*α; level = level - β;*
end if
Iterations= Iterations+1;
until *Iterations > N.iters && Temp < T_f (termination conditions are met)*
Step 3: Termination phase
Return the best found solution $S_{Arrange;}$
end Procedure

Figure 1. Pseudo code for hybridize modified great deluge algorithm with iterative simulated annealing for medical data clustering problems

In the improvement phase (Step-2), basically the initial solution is iteratively improved by employing the hybridization method (ISA-MGD) until the stopping condition is met. In Step-2.1, neighbourhood structures N1 and N2 are applied to generate candidate solutions (in this case, five candidate solutions are generated), and the best candidate solution is selected as $S_{working}$. There are two cases to be taken into account such as:

- Case 1: Better solution
  If $f(S_{working})$ is better than $f(S_{Arrange})$, then $S_{working}$ is accepted as a current solution ($S_{source} \leftarrow S_{working}$), the best

solution is updated ($S_{Arrange} \leftarrow S_{working}$), and the list value is updates (FIFO) by *level* (*LL*← *level*) as shown in Step-2.2. The *Temp* will be decreased by the value α (i.e. *Temp = Temp-Temp*α*) and the *level* will be updated by the value β (i.e. *level = level - β*).

- Case 2: Worse solution

  If $f(S_{working})$ is less than $f(S_{Arrange})$, then the difference between the quality of $S_{working}$ and S_{source} is calculated. A random number [0, 1], *RN*, is generated. If the probability (i.e. $e^{-\delta/Temp}$, where $\delta = f(S_{working})-f(S_{source})$) is less than or equal to *RN*) then $S_{working}$ is accepted, and the current solution is updated ($S_{source} \leftarrow S_{working}$). Otherwise, the quality of $S_{working}$ is compared against the *level*. If it is less than or equal to the *level*, then $S_{working}$ is accepted, and the current solution is updated ($S_{source} \leftarrow S_{working}$). Otherwise, $S_{working}$ will be rejected. The *level* will be updated by the value β (i.e. *level = level - β*). The counter for not improving solution is increased by 1. If this counter is equal *non_improving_length_GD*, then we reinitialize the temperature (*Temp*) equal to the initial temperature (T_o), and the *level* is updated by a new *level* that is randomly selected from the list (where the size of the list is set to 10 based on preliminary experiments). Otherwise, the process continues untill the stopping conditions are met (i.e. *Iterations> N.iters && Temp* < T_f), and return the best solution found $S_{Arrange}$ (Step-2*)*. Note that in this work the *est.q* is set to 0, and *non_improving_length_GD* is set to 10 (after some preliminary experiments).

The process will continue until the termination conditions are met (*Iterations> N.iters && Temp* < T_f), and return the best solution found so far $S_{Arrange}$ (Step-3).

IV. Results and Discussion

In this work, we ran our algorithm 20 times across 6 datasets. The algorithms are programmed in Java language and are tested on a PC with an Intel dual core 1800 MHz, 2GB RAM. In the analysis part, the terms used are as follows:

- *N-NI*: Number of not improved iterations.
- *N-I*: Number of improved iterations.
- *Std*: Standard Deviation.

The parameters setting used in these experiments are shown in Table I for ISA-MGD algorithm, where it shows that, the parameter setting for ISA such as, T_o, T_f and α , in addition to the parameters setting for MGD such as *est.q*, n*on_improving_length_GD*, and *N.iters*, in addition to *LL* parameter that stores the level values as in [13]

TABLE I. Parameters setting used in the hybridize iterative simulated annealing with the modified great deluge algorithm (ISA-MGD).

Parameter	Description	Value
T_o	Initial temperature	0
T_f	Final temperature	10
A	Cooling rate	0.7
N.iters	Number of iterations	100,000
est.q	Estimated quality value	0
non_improving_length_GD	Number of not improved solutions	600
LL	Size of list	10

Table II shows the results comparison between IISA, MGD and ISA-MGD algorithms based on the minimal distance calculation (i.e. between objects). The "Avg" represents the average results out of 20 runs. The best results are presented in bold. Table III shows the results comparison between IISA, MGD and ISA-MGD algorithms based on the minimal distance calculation (i.e. between Centers). The "Avg" represents the average results out of 20 runs. Again, the best results are presented in bold.

The results in Table II show that, ISA-MGD algorithm outperform MGD and IISA algorithms on all datasets except in one dataset for each based on the minimal distance calculation (i.e. between objects). Table II also indicates that, ISA-MGD algorithm using N1 and N2 neighborhood structures better than MGD and IISA algorithms using N1 and N2 based on best minimal distance and average score.

The results in Table III show that, ISA-MGD, MGD and IISA algorithms obtain same best result in some instances based on the minimal distance calculation (i.e. between centers). Table III also indicates that, ISA-MGD algorithm using N1 and N2 neighborhood structures better than MGD and IISA algorithms using N1 and N2 in some instances based on average score.

TABLE II. Results between Objects obtained from IISA, MGD and ISA-MGD algorithms using N1 and N2.

Dataset	Initial length by *K-Means*	20 Runs – Minimal Distance Calculated as between Objects					
		IISA		MGD		ISA-MGD	
		Best	Avg	Best	Avg	Best	Avg
1. B.C	6379.694	2338	2430.05	**1937**	2088.4	2090.61	2124.76
2. T.D	3178.714	1228.8	1375.23	893.33	946.69	**892.57**	929.28
3. B.L.D	17258.715	5771.59	6028.67	5509.21	5809.55	**5466.33**	5631.02
4. L.C	182.577	158.98	161.14	159.27	161.38	**157.8**	159.20
5. H.S	2463.972	**987.77**	1049.03	1023.96	1069.57	1014.05	1043.21
6. P.I.D	100880.390	24920.39	26038.86	23281.12	24239.71	**22919.27**	24118.18

TABLE III. Results between Centers obtained from IISA, MGD and ISA-MGD algorithms using N1 and N2.

Dataset	Initial length by *K-Means*	20 Runs – Minimal Distance Calculated as between Centers					
		IISA		MGD		ISA-MGD	
		Best	Avg	Best	Avg	Best	Avg
1. B.C	5360.710	**2778**	3104.4	3014.72	3316.57	3007.32	3281.79
2. T.D	2459.620	**2039.89**	2054.09	**2039.89**	2047.04	2070.16	2079.52
3. B.L.D	22646.89	**10498.9**	10855.71	**10498.9**	10836.97	**10498.9**	10969.26
4. L.C	168.520	152.37	154.28	**151.62**	153.49	152.37	153.04
5. H.S	3626.530	**2721.36**	2722	**2721.36**	2721.594	**2721.36**	2730.16
6. P.I.D	102398.583	**48909.2**	54751.22	**48909.2**	56159.74	**48909.2**	57098.7

Table IV shows a further analysis on ISA-MGD using N1 and N2 neighborhood structures between objects over all tested datasets. For example, the best results for *L.C* dataset is 157.8 that is obtained within 12 minutes 13 seconds under 22387 iterations. Meanwhile, the range for minimum and maximum results is in between 157.8 and 160.39. In most of the cases, the results are obtained between 12 minutes 13 seconds to 19 minutes 35 seconds that are considered acceptable.

For example, Fig. 2 shows a 3D scatter graph for *K-Means*, MGD and ISA-MGD algorithms using N1 and N2 neighborhood structures for the calculation between objects over *H.S* dataset. Two clusters are represented by the two colors. Fig. 2-a shows that, the two clusters (colors) are mixed with initial minimal distance obtained by K-Means is 2463.972. Whereas, Fig. 2-c shows that, ISA-MGD obtained better improvement in terms of the minimal distance than MGD algorithm which is equal to 1014.05.

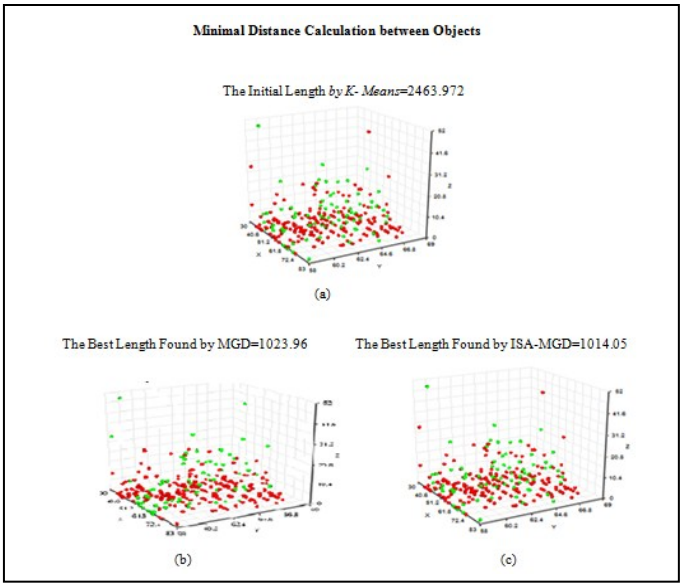

Figure 2. Scatter graph for K-Means, MGD and ISA-MGD algorithms over H.S dataset for minimal distance calculation between objects using N1 and N2 neighborhood structures

TABLE IV. Results analysis "between Centers" for MGD algorithm using N1 and N2 neighborhood structures on six datasets

Dataset Description	*B.C*	*T.D*	*B.L.D*	*L.C*	*H.S*	*P.I.D*
Best	2090.61	892.57427	5466.33	157.8	1014.05	22919.27
Iterations for best	98334	99731	98479	22387	99962	99921
Time	00:44:28	00:11:02	00:03:05	00:12:13	00:01:19	00:12:13
N-I	58941	60498	53598	14066	20227	62542
N-NI	38855	38871	45269	85878	79122	34918
Average	2124.7632	929.28045	5631.021	159.20720	1043.211	24118.18
Result Range	2090.61 2200.44	892.57427 967.54135	5466.33 5857.91	157.8 160.39	1014.05 1083.1	22919.27 24761.49
Iterations Range	98334 99999	89602 99936	95286 99928	18564 98736	93112 99962	99591 100001
Time Range	00:44:02 01:09:35	00:10:44 00:11:35	00:02:59 00:03:43	00:12:13 00:19:35	00:01:16 00:01:29	00:08:14 00:12:39
Std	31.357802	20.24604	103.55513	0.71663	21.87092	437.6589

V. CONCLUSION

This work propose hybridize between modified great deluge algorithm and iterative simulated annealing (ISA-MGD) for medical clustering problem, the idea of the hybridization is to diverse the search space, after that the performances of the proposed algorithm is compared based on the minimal distance that is calculated based on (i) between objects, and (ii) between centers. The algorithms has been implemented and tested on six well known real datasets. The algorithms in the comparison are our previous proposed modified great deluge algorithm (coded as MGD) and our previous proposed extend iterative simulated annealing (coded as IISA). Two different neighborhood structures are employed within the proposed approaches i.e. (N1 and N2).

Experimental results show that, ISA-MGD algorithm in the minimal distance as between objects calculation performs better compared to the MGD and IISA algorithms based on minimal distance and the average score. Where, ISA-MGD produce good quality solution compared to MGD and IISA algorithms in the minimal distance calculation as between centers.

Generally, it can be concluded that, the algorithms behave differently due to the different measurements imposed during the search process. However, the limitation in MGD, IISA, ISA-MGD algorithms is the neighbourhood structures i.e. N1 and N2 are not really effective as it is based at random, which pose a future work.

REFERENCES

[1] X. Wang, "Fuzzy Clustering in the Analysis of Fourier Transform Infrared Spectra for Cancer Diagnosis," PhD thesis School of Computer Science and Information Technology University of Nottingham, 2006.

[2] I. Saha, D. lewczyński, U. Maulik, and S. Bandyopadhyay, "Consensus Multi objective Differential Crisp Clustering for Categorical Data Analysis," RSCTC 2010, 2010; 30-39.

[3] P. Brucker, "On the complexity of clustering problems," In: Beckmenn. M., Kunzi, H.P. (eds.) Optimization and Operations Research. Lecture Notes in Economics and Mathematical Systems, Springer, Berlin vol. 157, pp. 45–54, 1987.

[4] S. Dasgupta and Y. Freund, "Random Projection Trees for Vector Quantization," Information Theory. IEEE Transactions on, vol. 55, issue 7, pp. 3229-3242, 2009.

[5] M. Mahajan, P. Nimbhorkar, and K. Varadarajan, "The Planar k-Means Problem is NP-Hard," Lecture Notes in Computer Science, WALCOM2009, Vol.543.: pp. 274-285, 2009.

[6] M. Berry and G. Linoff, "Handbook on Data Mining Techniques—For Marketingm," Sales and Customer Support. Publisher John Wiley & Sons, NY, USA, 1997.

[7] U. Maulik and S. Bandyopadhyay, "Genetic Algorithm-Based Clustering Technique," Pattern Recognition. vol. 33: pp. 1455-1465, 2000.

[8] J. L. Lin, and M. C. Wei, "Genetic algorithm-based clustering approach for k-anonymization," Jornal Expert Systems with Applications: An International Journal archive, Expert Syst. Appl. Pergamon Press, Inc. Tarrytown, NY, USA, vol 36 Issue 6, pp.9784-9792, 2009.

[9] B. Al-Shboul and S. H. Myaeng, "Initializing K-Means using Genetic Algorithms," World Academy of Science, Engineering and Technology 54, 2009.

[10] . Yongguo, Y. Zhang, W. Hong, Y. Mao, and C. Kefei, " A tabu search approach for the minimum sum-of-squares clustering problem," Information sciences, vol. 178, Issue 12, pp. 2680-2704, 2008.

[11] C. Zhang, D. Ouyang, and J., Ning, "An artificial bee colony approach for clustering," Expert Systems with Applications, vol 37, pp. 4761–4767, 2010.

[12] R. Kittaneh, S. Abdullah, and A. Abuhamdah, "Iterative Simulated Annealing for Medical Clustering Problems," Processing of the Trends in Applied Sciences Research Journal, DOI:10.3923, Jan, 2012.

[13] A. Abuhamdah, "Modified Great Deluge for Medical Clustering Problems," Accepted in The International Journal of Emerging Sciences.

[14] S. Hong, "Experiments with K-means, fuzzy C-means and approaches to choose K and C," PhD thesis University of Central Florida Orlando, Florida, 2006.

[15] P. Berkhin, "Survey of Clustering Data Mining Techniques," San Jose, CA, USA, Accrue Software. available from accrue.com, 2002.

[16] A, Jain and R. Dubes, "Handbook of Algorithms for Clustering Data," Prentice-Hall advanced reference series, Prentice-Hall. Englewood Cliffs, NJ, USA, 1988.

[17] G. Dueck, "New optimization heuristics: Great Deluge and the Record-to-Record Travel," Journal of Computational Physics, vol 104, issue 1, pp. 86–92, 1993.