A new P system with hybrid MDE-k-means algorithm for data clustering

WEISUN, LAISHENG XIANG, XIYU LIU
School of Management Science and Engineering
Shandong Normal University
Jinan, Shandong
CHINA
sdxyliu@163.com

Abstract:- Clustering is an important part of data mining. It can immensely simplify data complexity and helps discover the underlying patterns and knowledge from massive quantities data points. The popular efficient clustering algorithm k-means has been widely used in many fields. However, The k-means method also suffers from several drawbacks. It selects the initial cluster centers randomly that greatly influences the clustering performance. This study proposes a new P system with modified differential evolution -k-means algorithm to improve the quality of initial cluster centers of k-means algorithm. The P system has three types of membranes: elementary, local store, global store. Different membranes have different rules. Based on the membrane structure, the elementary membranes evolve the objects with modified differential evolution algorithm and other types of membrane update the local best and the global best objects synchronously with communication rules. Under the control of the P system, the hybrid algorithm achieves a good partition for data sets, compared with the classical k-means algorithm and DE-k-means algorithm.

Key-Words: - Data mining; clustering; unsupervised learning; k-means; modified DE algorithm; membrane computing.

1 Introduction

Clustering also known as unsupervised learning in general is the process of dividing n unlabeled data points into k labeled groups. As a result, the elements in the same cluster have higher similarities than those in different clusters. Over the years, a great number of clustering algorithms have been developed [9].

One of the most popular unsupervised clustering methods is the k-means algorithm [14]. This method has the advantages of simplicity, easy to implement and high efficiency. This method has been widely used in a variety of areas. However, the k-means algorithm also has some drawbacks. It may get struck at a local optimum and is very sensitive to the initial choice of the cluster centers. In order to overcome these drawbacks, various clustering algorithms based on evolutionary computing have been introduced in recent years, such as the generic algorithms, the differential evolution (DE) algorithm, and the particle swarm optimization algorithm [2, 4, 5, 7, 10, 11].

DE is undoubtedly an effective population based stochastic optimization algorithm [1]. DE algorithm is robust, reliable, inherently parallel, easy to implement and has great global exploration ability. It can achieve the global optimal solution as the initial cluster centers for the k-means algorithm and improve the clustering performance. However, DE also suffer some drawbacks. It has weak local exploitation ability and low convergence velocity.

This study proposes a modified DE algorithm to overcome these disadvantages.

Membrane computing, first introduced by Paun in 1998 [15] is motivated by the structure and function of a living cell. There are three main types of P systems, i.e., cell-like P systems, tissue-like P systems, and neutral-like P systems [6]. These P systems consist of original objects, special membrane structures and specific flexible evolution and communication rules. Membrane computing has the advantage of distributed and maximum parallelism and can greatly improve the computational efficiency. In this study, an innovative membrane structure is built and the inherent evolution-communication rules are modified. Through these operations, the global best object is eventually obtained.

2 The classical differential evolution algorithm

The DE algorithm, proposed by Storn and Price [13] and used to find global optimal solutions for hard optimization problems, has drawn more attention in recent years. The best setting of the control parameters can be different for different optimization problems. The DE algorithm mainly consists of parameters, the fitness function and three differential operations including mutation, crossover and selection.

Let i is the $i-th$ object in the population, X_i denote a genome in the population, M_i indicate the vector that has gone through the mutation operations and R_i be a new vector undergone crossover operations. Set V_i to represent the survival vector after a selection operation.

2.1 Mutation operation

The mutation scheme plays an important role in the process of searching for a global optimal solution and accelerating convergence. There are five forms of mutation operations as given by, and several studies have proved that the first and the fourth form have better performances.

1. DE/rand/1: $M_i = X_1 + F*(X_2 - X_3)$
2. DE/best/1: $M_i = X_b + F*(X_2 - X_3)$
3. DE/current-to-best/1:
$M_i = X_i + F*(X_b - X_1) + F*$（$X_2$ - X_3)
4. DE/best/2:
$M_i = X_b + F*(X_1 - X_2) + F*$（$X_3$ - X_4)
5. DE/rand/2:
$M_i = X_1 + F*(X_2 - X_3) + F*$（$X_4$ - X_5)

In the above, X_1 , X_2 , X_3 , X_4 , X_5 are random vectors in the population and are different from the base vector X_i , F is a positive predefined constant, called the scale factor for scaling different vectors, and X_b is the vector with the best fitness in the population.

2.2 Crossover operation

This operation can greatly increase the diversity of the population. The crossover operation is performed according to (1) in the following

$$\begin{cases} R_{ij} = M_{ij} & \text{if } rand[0,1] \le CR \text{ or } j = jrand \\ R_{ij} = X_{ij} & \text{otherwise} \end{cases}, \quad (1)$$

where CR denotes the crossover rate, $jrand$ ($jrand \in \{1,...,d\}$) is a randomly chosen index, R_{ij} is the j-th component of the data point R_i and M_{ij} is the j-th component of the newly generated data point M_i. This crossover operation ensures that object R_i gets at least one component from the newly generated vector.

2.3 Selection operation

Selection operation determines which vector to survive in the next generation based on the changes caused in the objective function. The objective function is also known as the fitness function. Different problem has different objective function and each of the survival vectors has a higher fitness. In this study, the selection operation is performed as follows:

$$f = \min(\sum_{i=1}^{d} \| X_{ij} - Z_{ij} \|^{2})$$

$$\begin{cases} V_i = R_i & \text{if } f(R_i) \le f(X_i) \\ V_i = X_i & \text{otherwise} \end{cases}, \qquad (2)$$

where f is the fitness function. $f(X_i)$ is the fitness of X_i.

3 The modified differential evolution algorithm

The DE algorithm has the disadvantages of weak local exploitation ability as well as low convergence velocity [3, 8].The modified DE algorithm overcomes these disadvantages. The new algorithm is called the modified differential evolution algorithm (MDE for short).

3.1 The modified of mutations operation

In the mutation operation, the classical DE algorithm chooses three vectors and the base vector randomly to maintain diversity, which slows down convergence. In this study a directed mutation scheme is proposed to enhance the local exploitation and to speed up convergence. In the MDE algorithm, the mutation operation is performed according to (3) and (4) in the following

$$\begin{cases} V_i^{G+1} = X_1^G + F*(\mathrm{X}_2^G \text{-} \mathrm{X}_3^G), & \text{if } rand(0,1) \ge \dfrac{G}{G_m} \\ V_{\mathrm{i}}^{G+1} = X_1^G + F_r*(X_g^G - X_w^G), & \text{otherwise} \end{cases} \qquad (3)$$

$$F_{\mathrm{r}} = 0.5*(1 + rand(0,1)), \qquad (4)$$

where G indicates the number of current iteration, G_m denotes the maximum number of iterations, V_i^{G+1} symbolizes the base vector of the next generation, X_g represents the data vector with the best fitness, X_w represents the worst vector and F_r is the dynamic scaling factor.

In (3) and (4), F is an important scaling factor that controls the evolving rate of different vectors. In general, F is a constant value in [0, 2]. In the directed mutation scheme, the base vector is maintained randomly and the different vectors are evolved following the direction of the best vector and the opposite direction of the worst vector. In order to increase the diversity of the population, F_r is treated as a random variable as determined by (4) to perturb the base vector.

The directed mutation scheme can explore the same direction but with different weights under certain conditions. At the beginning of the directed MDE, two mutation schemes can be used with a lager probability for the basic scheme and a smaller probability for the directed scheme. As more generations are performed, i.e., as G increases, $\frac{G}{G_m}$ also increases and the two schemes will be applied under the same possibilities. This strategy balances the exploration and exploitation of the MDE algorithm. Finally, it increases the probability of the directed scheme and enhance local exploitation. In the entire mutation process, the exploitation and the exploration are performed in parallel.

3.2 The modification of the crossover rate

In the crossover operation of the classical DE, CR controls the probability of the trial individual adopting the old individual genes or the new individual genes. More new genes will be adopted with a larger CR value. As a result, the population diversity increases and the convergence speeds up. However, a large CR value may also decrease the convergence rate and lead to a premature convergence result. At the start of the search, the population has better diversity and a small CR value is needed. The small value can help avoid converging prematurely. With the continuous development of evolution, a large CR value should be used to increase population diversity and accelerate the convergence.

In the MDE algorithm, the crossover rate CR is determined as follows:

$$\begin{cases} CR = CR_{\min} & \text{if } G = 0 \\ CR = CR_{\min} + (CR_{\max} - CR_{\min})(\frac{G}{G_m})^p & \text{otherwise} \end{cases}, \quad (5)$$

where $CR_{\min}$=0.1, $CR_{\max}$=1 and $p = 4$.

4 The P system with hybrid algorithm MDE-k-means

In this section, the P system is introduced in details. It includes the special membrane structure, the membrane rules, the calculating process and the halt condition in P system.

4.1 A brief introduction of the k-means algorithm

The core of the k-means algorithm is the measure of similarity between vectors based on the Euclidean distance. The k-means algorithm can be briefly stated as follows:

(1) the k-means algorithm randomly chooses the k cluster centers.

(2) each vector is assigned to the cluster with the shortest Euclidean distance between the vector and the cluster center vector among all k clusters. The cluster center vector is recalculated after a vector is assigned to the cluster.

(3) the algorithm terminates when the number of iterations exceeds the maximum number or when the changes in center vectors stays within a certain amount.

A new P system is designed in this study using a hybrid algorithm. This hybrid algorithm combines the MDE and the k-means algorithms (MDE-k-means for short) to optimize the cluster centers in the k-means algorithm so as to improve the clustering results.

4.2 The structure of the P system using the hybrid algorithm

The MDE-k-means P system has a nested structure of three layers with (2q+2) membranes.

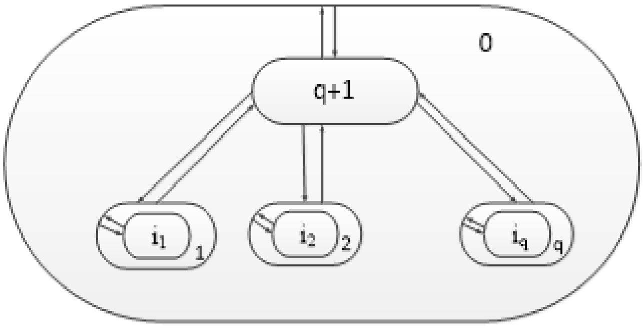


Fig.1 The membrane structure of P system with MDE-k-means algorithm

The new P system for clustering is defined as follows:

$$\prod = (O, \mu, M_{i_1}...M_{i_q}, R_{i_1}...R_{i_q}, R_0...R_{q+1}, i_0) \quad (6)$$

The notation in (6) are

O represents the alphabet of the objects in the P system. Each object is a $k*d$ dimensional vector.

μ represents the membrane structure of the P system.

$M_{i_1},...,M_{i_q}$ mean the initial objects on the membrane i_l (l=1...q).

$R_{i_1},...,R_{i_q}$ symbolize the evolution and the communication rules contained in the membrane

i_l (l =1...q).

$R_0,...,R_{q+1}$ denote the communication rules on the skin membrane and membrane labeled $0,...,q+1$.

i_0 indicates the output region. The output region is the environment if $i_0 = 0$.

4.3 The rules in the P system

The rules in the P system are introduced in this section.

4.3.1 evolution rules

The evolution rules on the elementary membrane consist of mutation rules (3, 4), crossover rules (1, 5) and selection rules (2) with the form $\mu \to \upsilon$ where $\mu, \ \nu \in O$.

4.3.2 communication rules

Communication rules mean membrane choose object with the highest fitness to transmit to their neighbor. In each iteration, elementary membrane communicate their local best object with their upper neighbor, which labeled $1,...,q$. Then Membranes labeled $1,...,q$ select the object with highest fitness on their own membrane and transfer the object to membrane $q+1$. Membrane $q+1$ can receive the local best object from $1,...,q$ and select the global best object from them to pass into the skin membrane.

4.3.3 k -means rules

$$D_{(x_i, z_j} = \sqrt{\sum_{m=1}^{d}(X_{im} - Z_{jm})^2} \qquad (7)$$

In formula (7), $X_i (i = 1,...,n)$ symbolizes the i -th vector in the data sets, Z_j denotes the center vector of the cluster $j(j = 1,...,k)$, and d means the features of each vector.

After each iteration, recalculate the cluster vectors.

$$Z_j = \frac{\sum_{x_i \in c_j} x_i}{n_j}, \qquad (8)$$

Where n_j is the total number of the data in cluster j. C_j means the cluster j.

4.4 The calculating process in the P system

The P system has a three layer nested structure. The outermost membrane, named skin membrane, is labeled 0 and contains $2q+1$ membranes. Each of the membranes $l(l = 1,...,q)$ have an elementary membrane labeled $i_l (l = 1,...,q)$.

Before the calculation starts, the elementary membrane $i_l (l = 1,...,q)$ generates m initial objects while other membranes do not have initial object. Each of the m objects is consist of $k*d$ random real numbers that generate randomly. They all satisfy the conditions that:

$$Z_{ij} = rand*(\max_j - \min_j) + \min_j, \qquad (9)$$

Where $\max_j$ and $\min_j$ are the upper and lower bounds on the j -th component of Z_i.

There are evolution rules that consist of selection rules, crossover rules, mutation rules and communication rules on the elementary membrane. In each iteration, elementary membrane pass their local best object into their higher neighbor. Membranes $1,...,q$ without initial object only have the communicate rules. They store the local best objects that passed from its elementary membrane, select the one with highest fitness and communicate with membrane $q+1$. Membrane $q+1$ store the local best objects from $1,...,q$ and select the global best object from them to communicate with the skin membrane.

4.5 Halting and output of the P system

In the P system, each membrane is considered as a parallel computing unit with high efficiency. In every operation the rules on the same membrane are applied in a maximally parallel manner until it halts.The P system halts, When the execution steps arrive the maximum numbers. And the object stored in the skin membrane is regarded as the best cluster centers.

4.6 Complexity analysis

Considering the total number of records is n; The number of attributes is d; every membrane has m initial records; the number of iterations of crossover, mutation and selection is N; The detailed complexity analysis is presented as follows.

Step 1: Normalize a data set

In a data set, normalize the values of numerical attribute. The complexity to find the maximum or the minimum domain value of a numerical attribute is O(n). The complexity to normalize all the values of a numerical attribute is O(n).

Step 2: Initial population

There are m objects in every membranes and every object have d attribute. Complexity of record to record distance. The complexity to calculate distance between two records is O(d). For a membrane with m records, the complexity for record to record distance calculation is O(m^2d).

Step 3: Selection operation

It computes the distance between all pairs of seeds. If there are k number of seeds the complexity is O(k^2d). The fitness function also computes the distance between each record and its closest seed with a complexity O(mkd). Therefore, the complexity of the fitness calculation for each record is O(mkd+k^2d).

Once the fitness values of the m records are computed we need to sort them in descending order for finding the best records. The complexity for this is O(m^2). Therefore the total complexity of selection operation is O(mkd+k^2d+m^2).

Step 4: Crossover operation

In the crossover operation, The rearrangement complexity is O(k^2d) if there are k genes in a chromosome and d attributes in the data set. for m chromosomes the complexity of twin removal is O(k^2dm).

Step 5: Mutation operation

If a chromosome is chosen for mutation, the complexity of this is O(k),the number of gene in a chromosome is k. In the mutation operation we need to compute the maximum fitness and the minimum fitness for calculating the mutation probability. The complexity for the calculation is O(m) if there are m records in one membrane. Therefore, the complexity of the Mutation operation is O(km).

If there are N iterations then the Selection operation, Crossover operation, Mutation operation will be repeated N times while the initial population will be selected once randomly.

Step 6: k -means

If the number of iterations in k -means is the $N^{'}$， then the complexity for k -means is O(nkdN’).

Therefore, the overall complexity of MDE-K-means. For a high dimensional data set where d is very large compared to all other parameters the complexity of MDE-K-means is O(d). Similarly, the overall complexity of MDE-K-means for a very large data set where the number of records n is very large compared to all other parameters the complexity is O(n).

5 Experimental results

This section mainly introduced the experiments, which included the data sets and parameters in experiments, the process of experiment and the experiment results.

5.1 Experiment data sets and parameters

In the paper, we evaluate and compare the P system with modified differential evolution- k -means algorithm with the classical k-means algorithm and the differential evolution-k-means algorithm. Experimental results were calculated for artificial data sets, the iris data set and the wine data set from UCI Machine Learning Repository.

For the artificial data set, we calculate the compactness of a cluster and the separation between the clusters to evaluate the clustering performance of the algorithms.

$$comp_j = \frac{\sum_{X_i \in C_j} \| X_i - Z_j \|^2}{|C_j|} \tag{10}$$

$$sep_j = \min_{i \neq j} \| Z_i - Z_j \| \tag{11}$$

Where ||.|| means Euclidean distance between the two vectors. $|C_j|$ is the number of the vectors in cluster j. Z_i is the center of cluster i.

According to (10, 11), it is easily to find that a smaller compactness and lager separation imply a better clustering solution.

For the iris and the wine data sets, to evaluate the performance of the modified clustering algorithm with membrane computing, it is necessary to use a statistical-mathematical function, the cluster validity index, to measure the effectiveness of the proposed novel solution. We use the Xie-Beni index, the PBMF index, the clustering quality described below, in our experiment to judge the performance of these methods. These different types of data points make us do more experiments to prove the effectiveness of the new algorithm.

The Xie-Beni index is a function of the ratio of total variation σ, to minimum separation, sep, of the cluster.

The XB index is

$$XB = \frac{\sigma}{n \times sep(Z)} \tag{12}$$

In (12), $\sigma = \sum_{i=1}^{k} \sum_{j=1}^{n} u_{ij}^2 \| X_j - z_i \|^2$ (13)

$$sep(Z) = \min_{i \neq j} \| z_i - z_j \|^2,$$

In (13), $u_{ij} = \dfrac{(\frac{1}{\| X_j - z_i \|})}{\sum_{i=1}^{K} (\frac{1}{\| X_j - z_i \|})}$

In these formula, X_j means the $j\text{−}th$ unlabeled data points. The z_i represents the center points of the cluster C_i. The u_{ij} is the fuzzy membership degree of the X_j to C_i. Note that a better partitioning should has lower σ and higher sep. Therefore lower values of XB index achieve a good clustering solution.

The PBMF index

$$PBMF(K) = (\frac{1}{K} \times \frac{E_1}{E_k} \times D_K)^2 \tag{14}$$

In (14), $E_K = \sum_{i=1}^{K} \sum_{j=1}^{n} u_{ij} \| X_j - z_i \|$,

$$D_K = \max_{i \neq j} \| z_i - z_j \|.$$

K is the number of clusters. E_1 is a constant for a given data set. In usually larger PBMF implies a better partitioning.

The novel proposed MDE-k-means with P system parameters were set to: the cluster number k=3, scaling factor F=0.7, the number of initial objects on elementary membrane m=20, the number of generations G_m=100, crossover rate $CR_{\min}$=0.1, $CR_{\max}$=1, membrane numbers q=10.

5.2 Experiment process

Input: Data set, X; the number of clusters, K; the number of elementary membranes, i_q; the number of objects in each evolution membrane, m; maximum generation, G_m; and crossover probability, $CR_{\max}$, $CR_{\min}$; scaling factors F.

Output: The optimal cluster centers, O_{gbest}, in the skin membrane.

Begin

Step1: ∕ Initialization ∕

```
for i = 1 to $i_q$
    for j = 1 to m
        Generate j-th initial object for elementary membrane $i_l$, $O_j^{i_l}$;
        Calculate partition matrix, $Z_j^{i_l}$;
    end for
    Update local best object, $O_{\text{ilbest}}$, in i-th local store membrane by communication rules;
end for
Update global best object, $O_{\text{gbest}}$, in global store membrane by communication rules;
Set Generation G = 0;
Step2: ∕ Object evolution in elementary membranes ∕
for each elementary membrane, $i_l$, ( l=1,   ,q) in parallel do
    for j = 1 to m
        Evolve objects, $O_j^{i_l}$, using the modified differential evolution rules;
        Calculate partition matrix, $Z_j^{i_l}$;
    end for
end for
Step3: ∕ Object communication ∕
for each elementary membrane, $i_l$;( l=1,   ,q), in parallel do
    Transmit the best object in membrane $i_l$ to update its local best object, $O_{\text{ilbest}}$, using communication rules;
    Transmit the best object in membrane i to q+1 to update the global best object, $O_{\text{gbest}}$, using communication rules;
end for
step4: ∕ Halting condition judgment ∕
if $G \le G_m$ is satisfied
    $G = G + 1$;
    go to Step2;
end if;
Export the global best object, $O_{\text{gbest}}$;
Calculate partition matrix, $Z_j$, according to the global best object, $O_{\text{gbest}}$;
Assign all points into K clusters based on $Z_j$.
End;
```

5.3 Experiment results

For artificial data set: X=[10 15;27 21;5 9;72 69;96 78;34 39;55 72;39 46;22 35;71 77;23 29;35 39;41 54;22 16;35 62]

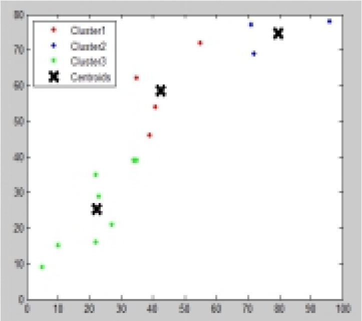
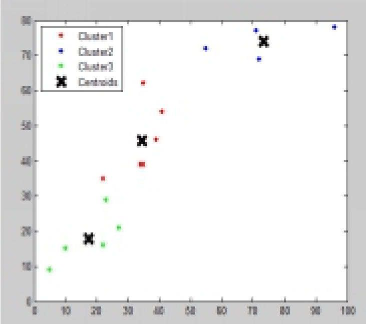
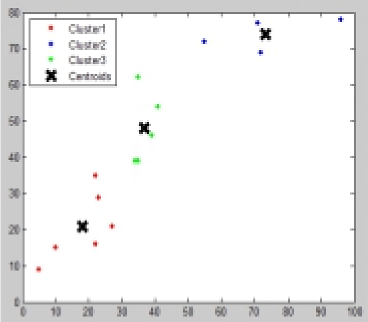
k -means	
DE- k -means	
MDE- k -means	

Fig.2 The clustering result with three methods for artificial data sets.

Table1 The average compactness of the three methods in clustering artificial data sets.

cluster	k -means	DE- k -means	MDE- k -means
1	139.2778	134.5	126.3613
2	227.75	216.4252	167.5765
3	115.44	149.775	86.96

Table2 The average separation of the three methods in clustering artificial data sets.

cluster	Average Separation of k -means	Average Separation of DE- k -means	Average Separation of MDE- k -means
1	1.0852	1.2736	1.5073
2			
3			

Table3 The average performance of the three methods in clustering UCI data sets.

method	Iris		Wine	
	Correct points	Correct rates	Correct points	Correc t rates
k -means	135	90%	118	66.3%
DE- k -means	141	94%	126	70.8%
MDE- k -means	146	97.3%	146	82.1%

Table 4 the XB and PBMF index of the three methods in clustering UCI data sets.

Data sets	Methods	XB	PBMF
Iris	k -means	0.946	6.53
	DE- k -means	0.534	9.65
	MDE- k -means	0.102	11.69
Wine	k -means	1.237	25368.34
	DE- k -means	0.425	25423.56
	MDE- k -means	0.096	25456.89

Through these experimental results, the conclusion could be drawn that MDE- k -means algorithm with P system has smaller compactness values and lager separation values compared with k-means method and DE- k -means. MDE- k -means algorithm with P system also has higher average correct rates. Contrast with other two method, MDE- k -means has the lowest XB values and the highest PBMF values. In a word, MDE- k -means algorithm with P system gets a better clustering partition and greatly shorten the computing time.

6 Conclusions

This paper presents a new P system with special membrane structure and modified evolution-communication mechanism. The P system combined with modified differential evolution- k -means algorithm can find the global optimal cluster centers to improve the performance of k -means algorithm. The initial objects on elementary membrane generate randomly and execute the evolution-communication rules in the P system. Under the control of rules, the initial objects are evolved and transmitted in the P system. Membranes update the local best objects and the global best object synchronously until system reach the halt conditions .In the end ,the global best object is the best approximation to optimal cluster centers for the k -means algorithm. The experimental results verify the advantages of the proposed P system with the hybrid algorithm.

Acknowledgement

Project supported by National Natural Science Foundation of China (61170038,61472231), Jinan City independent innovation plan project in College and Universities, China (201401202), Ministry of education of Humanities and social science research project, China (12YJA630152), Social Science Fund Project of Shandong Province, China (11CGLJ22), outstanding youth scientist foundation project of Shandong Province, China (ZR2011FM001).

References:

[1] Peng, H., Wang, J., Pérez-Jiménez, M. J., Riscos-Nunez, A. An unsupervised learning algorithm for membrane computing. *Information Sciences*,Vol.304, 2015,pp.80-91;.

[2] Kao Y T, Zahara E, Kao I W. A hybridized approach to data clustering. Expert Systems with Applications An International Journal, Vol.34,No.3,2008, pp.1754-1762.

[3] Mohamed A W, Sabry H Z, Khorshid M. An alternative differential evolution algorithm for global optimization. Journal of Advanced Research,Vol.3,No.2, 2012, pp.149-165.

[4] Rahman M A, Islam M Z. A hybrid clustering technique combining a novel genetic algorithm with K-Means. Knowledge-Based Systems, Vol.71,No.71, 2014, pp.345-365.

[5] Chang D X, Zhang X D, Zheng C W. A genetic algorithm with gene rearrangement for K-means clustering. Pattern Recognition, Vol.42,No.7, 2009, pp.1210-1222.

[6] Paun G. Computing with Membranes. Journal of Computer & System Sciences, Vol.61,No.1, 1998, pp.108-143.

[7] Bandyopdhyay S, Maulik U. An evolutionary technique based on K-Means algorithm for optimal clustering in RN. Information Sciences, Vol.146,No.2, 2002, pp.221-237.

[8] Noman N, Iba H. Accelerating Differential Evolution Using an Adaptive Local Search. Evolutionary Computation IEEE Transactions on, Vol.12,No.1, 2008, pp.107-125.

[9] Anil K. Jain. Data clustering: 50 years beyond K-means. Pattern Recognition Letters, Vol.31, 2010, pp.651-666.

[10] Tan S C, Kai M T, Teng S W. A general stochastic clustering method for automatic cluster discovery. Pattern Recognition, Vol.44,No.10, 2011, pp.2786-2799.

[11] Maulik U, Sanghamitra B. Genetic algorithm-based clustering technique. Pattern Recognition, Vol.33,No.99, 2000, pp.1455-1465.

[12] Chang D X, Zhang X D, Zheng C W. A genetic algorithm with gene rearrangement for K-means clustering. Pattern Recognition, Vol.42,No.7, 2009, pp.1210-1222.

[13] Rainer Storn , Price K. Differential Evolution – A Simple and Efficient Heuristic for global Optimization over Continuous Spaces. Journal of Global Optimization, Vol.11,No.4, 1997, pp.341-359.

[14] Anil K. Jain. Data clustering: 50 years beyond K-means. Pattern Recognition Letters, Vol.31, 2010, pp.651-666.

[15] Paun G. Membrane Computing[M]. Fundamentals of Computation Theory. Springer Berlin Heidelberg, 2003.