An Approach to Reducing Overfitting in FCM with Evolutionary Optimization

Seyed Mahmood Hashemi*

School of Computer Engineering, Darolfonoon High Educational Institute, Qazvin, Iran

Abstract

Fuzzy clustering methods are conveniently employed in constructing a fuzzy model of a system, but they need to tune some parameters. In this research, FCM is chosen for fuzzy clustering. Parameters such as the number of clusters and the value of fuzzifier significantly influence the extent of generalization of the fuzzy model. These two parameters require tuning to reduce the overfitting in the fuzzy model. Two new cost functions are developed to set the parameters of FCM algorithm properly and the two evolutionary optimization algorithms, i.e. the multi-objective simulated annealing and the multi-objective imperialist competitive algorithm, are employed to optimize the parameters of FCM according to the proposed cost functions. The multi-objective imperialist competitive algorithm is the proposed algorithm.

Keywords: Overfitting, fuzzy system modeling, FCM, multi-objective optimization

1. Introduction

The model of a nonlinear system can be obtained mainly by two methods: a global model can be obtained by finding and understanding the common relationship between all inputs and outputs of the system, and a local model of the system is obtained by describing input-output relations for a set of regions of input space by a simple function [1]. The relations in each partition of the input space can be expressed by a rule. Fuzzy Logic is suitable to describe the input-output relations by fuzzy rules with smooth transitions between the partitions [24, 25]. Fuzzy logic is preferred because it can combine linguistic models with qualitative models [2]. The two main processes in constructing a fuzzy model are the structure identification of the system and the parameter identification for the fuzzy model. Structure identification involves selection of suitable inputs and output and selection of type and number of fuzzy rules which splits the input space into regions. Parameter identification is involved in describing membership functions of the fuzzy rules. Some methods are proposed for structural and parameter identification of fuzzy models, where fuzzy clustering can do identification process in a rapid-prototyping approach [1, 21, 22]. The unsupervised learning ability of Fuzzy C-Means (FCM) clustering method and several kinds of cluster validity indices provide simple methods for approximate structural and parameter identification of the fuzzy models [3, 23, 20, 1]. Bezdek's FCM partitions a data set into C fuzzy clusters according to their similarities to the fuzzy mean of each cluster by a distance measure defined on data vectors [3]. So after vectors are scored for each cluster by their distance any fuzzy inference engine can be applied [1, 26]. Inference of clusters is based on the distance between input vector and centers of clusters; so determination of number of clusters is an important issue. Methods of determination of proper number of clusters were already proposed by some researchers based on various criteria [1, 2, 3, 5, 6, 7, 8].

The proposed algorithm in this paper modified the criteria of structural parameter determination to reduce overfitting. The validity of the proposed criteriais tested using Evolutionary Multi Objective Optimization Algorithm (EMOOA) in determining the optimum structural parameters for FCM based system modeling.

This paper is organized as follows: Section 2 explains fuzzy sets and modeling. Section 3 explains FCM algorithm and importance of the parameters in FCM

* Corresponding author. Email: hashemi2138@yahoo.com

algorithm. Section 4 explains evolutionary multi-objective optimization and two kinds of these algorithms (simulated annealing and imperialist competitive algorithm). Section 5 presents the proposed algorithm to improve FCM in reducing the overfitting. In Section 6, some metrics for comparingthe two aforementioned evolutionary algorithms are presented and experimental results of each evolutionary algorithm are presented.

2. Fuzzy Sets and Modeling

A Fuzzy Set is characterized by a membership function which associates with each point in space of points a real number between 0 and 1 [24]. In other words, each point x in a fuzzy set A is represented by a value between zero and one and this value explaind how much point x belongs to A [25].

Let a system with uncertainty have the input output relation $y = f_S(x)$, where $y \in R$, and $y \in R^{nX}$. A fuzzy system represents the knowledge related to inputs and output by nC fuzzy rules R_1, …R_C which are expressed in the form

R_i : If $(x_{k,1}$ isr$A_{i,1})$ and … and $(x_{k,nX}$isr$A_{i,nX})$ then $(y^*_{k,i}$ isr$B_i)$.

Where $y_k=f_S(x_k)$ is an observation vector (x_k, y_k) of the system; $x_{k,j}$ is the j^{th} variable of x_k; $A_{i,j}$ is the membership function of the fuzzy set for the j^{th} variable in the i'th rule, which determines a fuzzy number for the j'th variable of input space; $y^*_{k,i}$ is the estimate of $y_k=f_s(x_k)$ by R_i; the operator "and" denotes the *t-norm* operation between two membership values; and "isr" denotes the belonging of an object into a fuzzy set.

Fuzzy logic is a suitable medium for modelling and when the model becomes complete, it can produce estimates of output for an input outside the training data. This ability is named as *generalization* and disability of system for this aspect is called *overfitting*. In other words, overfitting means the model approximates the system by available data correctly, but it is not able to produce proper results for verification data [18, 19]. Reducing the overfitting is based on tuning the parameter of system, and, the most significant two parameters in fuzzy clustering (FCM) are number of clusters C and value of fuzzifier (m). The aim of this paper is proposing a method to deal this problem.

3. FCM Algorithm

The FCM algorithm scores each data vector $x_i = (x_{i,1}, \dots, x_{i,k}) \in R^k$ in the data set $\{x_1, x_2, \dots, x_N\}$ into C clusters according to a distance measured by solving the cost function [6]:

$$\begin{aligned} &\min J_m(U, V) = (U, V)^m \, dist^2(x_i, v_a) \\ &u_{a,i} \in [0, 1]; \ \forall\, a = 1, \dots, C; \ \forall\, i \in I \\ &\sum_{a=1}^{C} U_{a,i} = 1; \ 0 < \sum_{i=1}^{N} U_{a,i} < 1 \end{aligned} \tag{1}$$

where $U=(u_{a,i}) \in R^{C.N}$ is the partition matrix, also called the fuzzy-membership matrix; $V = (v_{a,k}) \in R^{C.k}$ is the matrix of cluster centers, v_a is the center of a^{th} cluster; dist(x_i, v_a) is the distance between vectors x_i and v_a. The scalar $m>1$ is called *fuzzifier* or *fuzzification power* , and it determines the fuzziness of clustering. If m is closer to 1 then $U_{a,i}$ tends to crisp values {0, 1}, and, if m is large then $U_{a,i}$ tends to distribute gradually in interval [0, 1].

$$u_{i,a} = \begin{cases} \left(\sum_{j=1}^{C} \left(\dfrac{\| x_a - v_i \|}{\| x_a - v_j \|} \right)^{2/(m-1)} \right)^{-1} & ; if \ \| x_a - v_j \| > 0 \\ 1 & ; if \ \| x_a - v_i \| > 0 \\ 0 & ; if \ \exists j \neq i, \| x_a - v_j \| = 0 \end{cases} \tag{2}$$

where$a$= 1, …, N and i= 1, …, C.

$$v_i = \frac{\sum_{a=1}^{N} u_{i,a}^m x_a}{\sum_{a=1}^{N} u_{i,a}^m} \tag{3}$$

Mostly Euclidian distance is preferred in clustering real data sets:

$$\text{dist}\left(X_a, V_i\right) = \left[\sum_{s=1}^{K} \left(x_{a,s} - v_{i,s} \right)^2 \right]^{1/2} \tag{4}$$

Getting the optimum solution for (1) is difficult [6]. A deterministic algorithm is proposed by some researchers [15, 16] to solve this optimization problem, which might fail to get the global optimum. An alternative solution for FCM algorithm is defined by [5]. Some researchers propose a method to specify the appropriate number of clusters [7, 8, 9]. In [2], it is proposed that clusters will provide the following two features: *minimum inside variance* (variance of vectors in that cluster) and *maximum outside variance* (variance between clusters). Another concept to be satisfied by the clusters is maximization of the average of membership values. The average of membership values is calculated through dividing sum of membership values of all data in a cluster by the number of data in that cluster [5]. Clusters which have low average of membership value are merged to the clusters to obtain higher average membership values. For this reason, authors proposed a formula for scoring the clusters:

$$S_i = \frac{\sum_{a=1}^{N} U_{i,a}}{N} \tag{5}$$

In [17], these aspects are stated in other words and it is said that *optimal partition* of data into subgroups were based on three requirements: (i) clear separation between resulting clusters; (ii) Minimal volume of clusters; (iii) Maximum number of data points concentrated in the vicinity of the cluster centroid. These aspects are defined based on the concept of *partitiondensity* which is defined by:

$$P_D = \frac{S}{\sum_{k=1}^{C}\left[\det\left(F_k\right)\right]^{1/2}} \tag{6}$$

where $S = \sum_{j=1}^{N}\sum_{k=1}^{C} U_{i,k}$,

$$F_k = \frac{\sum_{j=1}^{N} h\left(k \mid X_j\right)\left(X_j - V_k\right)\left(X_j - V_k\right)^T}{\sum_{j=1}^{N} h\left(k \mid X_j\right)} \tag{7}$$

and$h(k \mid X_j)$ is the probability of selecting the i^{th} cluster given the j^{th} feature vector. The average partition density is calculated from:

$$D_{PA} = \frac{1}{C}\sum_{i=1}^{C} \frac{S_i}{\sum_{k=1}^{C}\left[\det\left(F_k\right)\right]^{1/2}} \tag{8}$$

where $S_i = \sum_{j=1}^{N} U_{i,j}$.

In [9], these aspects are redefined as similarity and dissimilarity between clusters.

Overfitting occurs when clusters do not contain sufficient data vectors. In other words, inappropriate number of clusters cause overfitting; so for reducing the overfitting, an adaptation number of clusters is required. The process for adaptation number of clusters must follow conditions and formulas as stated above.

4. Evolutionary Multi-Objective Optimization

Multi-objective optimization algorithms (MOO) are necessary when multiple cost functions are considered in the same problem. The aim of MOO is tuning the decision variables to satisfy all objective functions F_i to optimum value. This class of problem is modelled by [10]

$$\begin{aligned} &\text{Optimize}[F_1(X),\cdots,F_k(X)] \\ &\text{Subject to: } g_i(X) \le 0,\ h_j(X)=0;\ i=1,\cdots,m\ ;j=1,\cdots,p; \end{aligned} \tag{9}$$

where$k$ is the number of objective functions, X is the decision vector, m is number of inequality constraints and p is number of equality constraints.

This goal causes differences between these algorithms and their ancestor single-objective optimization, which is based on concept of *best*, while the multi-objective optimization uses the concept of *dominance*. Dominance is defined in [10]:

$$\begin{aligned} &\vec{U} = \left(u_1,\cdots,u_k\right) \prec \vec{V}\left(v_1,\cdots,v_k\right) \\ &\quad \text{iff } \forall i, \{1,\cdots,k\}; u_i \le v_i, \exists j\ \{1,\cdots,k\} u_j < v_j \end{aligned} \tag{10}$$

In other words, a vector $\vec{U}$ of variables dominates another vector of variables $\vec{V}$ if and only if $\vec{U}$ can reach the optimal value for some criteria without causing a simultaneous non-optimal value for at least one criterion. If two vectors cannot dominate each other, they are called *non-dominated* vectors.

Many researchers proposed approaches for solving multi-objective problems by evolutionary methods [10, 11, 12, 13]. In this paper, multi-objective imperialist competitive algorithm (MOICA) and multi-objective simulated annealing (AMOSA) are considered.

A. Imperialist Competitive Algorithm (ICA)

Imperialist Competitive Algorithm (ICA) starts with an initial population (random solutions) which is called *countries* [28, 29, 30, 31]. Colonies are partitioned to *empires*. The cost or fitness of a country represents its *power*. In each empire, the country with the highest power is called *imperial*, and all remaining countries are called *colonies*. The total power of the empire is the sum of the powers of all countries in that empire. At each generation, the power of the countries is evaluated, and the imperial of an empire may change if one of its colonies enrich a power higher than the imperial. Each imperial absorbs colonies of other empires based on its power until the termination criterion of algorithm is satisfied. This imperialist competition results in the best collection of countries, which corresponds to a solution for a single-objective problem.

B. Multi-Objective Imperialist Competitive Algorithm (MOICA)

Multi-objective ICA requires keeping the non-dominated solutions in a list which is called *archive*. There are two important points (i) the algorithm must keep all non-dominated solutions in an *archive* to preserve from diversity. (ii) the algorithm must avoid deterministic methods to discover large number of non-dominated solutions. Instead of deterministic movements, using random movements helps the algorithm to escape from *local optimum* and increase the chance of reaching the *global optimum*. Therefore, it is better to construct the next population randomly from the archive or from the current population by a probabilistic method.

The first step is generation of population. Let N_{empire} be the initial number of empires, which may be changed during the process; $N_{country}$ be the number of countries and N_{values} be the number of values for each country. The population of of size (N_{empire} .$N_{country}$ .N_{values}) is randomly generated in a 3-dimensional matrix.

The second step in MOICA is evaluation of countries. The cost of each country is calculated using the value of that country according to an *objective function*; so if there are N_{object} objective functions in the problem and k^{th} value of country is presented by $value_k$ and cost of j^{th} country for i^{th}object function is represented by $cost^i_j$, the total power of j^{th} country is an array and calculated by:

$$power_j = cost^1_j(value_1,\ldots,value_n) + \ldots + cost^1_j(value_1,\ldots,value_n) \quad (11)$$

In the third step, the state of each country (colony or imperial) is decided. Countries are partitioned into $N_{empires}$ regions. In each empire, countries which are dominated (according to their power) by the other one are set as colony. The method for selecting imperials is randomly and probability for wining of each country in this process is based on *dominance count*. A non-dominated country is selected as imperial by probability

$$P(\text{imperial}) = S_{count} / S_{country}, \quad (12)$$

Where, S_{count} isthe number of countries which is dominated by candidate imperial and $S_{country}$ is the total number of countries in the empire. The remaining countries in the empire are set as colonies. If there is no non-dominated country in an empire, all countries become a colony. After that, the powers of all colonies are added to power of imperials, with the probability of imperial. A copy of the non-dominating imperials of each empire is kept in the archive.

In the fourth step, all colonies in all empires move toward the power of their imperials. For this assimilation policy, some random values (equal to N_{values}) between {0,1} are selected and the multiplication of these numbers to difference between power of all countries and power of their imperials is added to their latest power.

In the fifth step, the power of countries is re-evaluated and then the imperials are selected once again by the same procedure explained at the previous step. According to MOICA policy, all empires try to take the possession of colonies of other empires and control them. The weakest of all countries is colonized by the most powerful imperial. The number of countries which dominates a particular country determines the weakest colony, which is colonized by the most powerful imperial. The most powerful imperial is decided according to number of countries it can dominate.

C. Multi-Objective Simulated Annealing (AMOSA)

The basic concept in Simulated Annealing is evolution of the solution by simulating the decreasing temperature (*tmp*) in a material, where the higher the temperature means the higher the modification of the solution at a generation. If the temperature of a hot material decreases very fast, its internal structure may diverse and the material becomes hard and fragile. Decreasing the temperature slowly yields higher homogeneity and a less fragile material. Evolution of the solution is carried at specific temperature profiles. At the first iterations, a diverse set of initial solutions for the problem is produced at higher temperatures. And these solutions are evolved while the temperature decreases to get their local optimums. In multi-objective situation, there are non-dominated solutions which must be kept in the archive, as a candidate of optimal solution.

Along the runs of AMOSA algorithm, there are two solutions: current-so and new-so. They can have one of three states compared to each other: i- current-so dominates new-so, ii- current-so and new-so are non-dominated each other and iii- new-so dominates current-so.

If new-so is dominated by current-so, there may be solutions in the archive which dominates new-so. New-so is accepted to the archive by the probability

$$p = \frac{1}{1+\exp(\Delta \,.\, tmp)} \quad (13)$$

Where Δ is differencing between new-so and other solutions which dominate new-so

$$\Delta = \frac{\sum_i^k \Delta_i + \Delta}{k+1} \quad (14)$$

Solutions can escape from local-optima and reach to the neighbour hood of the global-optima by this probable acceptance.

If new-so is dominated by some solutions in the archive, (14) is modified to:

$$\Delta = \frac{\sum_{i=1}^k \Delta_i}{k+1} \quad (15)$$

When new-so is non-dominated with all members in archive, then new-so is set as current-so and it is added to the archive.

If new-so dominates some solutions in the archive, then new-so is set as current-so and it is added to the archive and solutions in the archive which are dominated by new-so are removed.

If new-so is dominated by some solutions in the archive, then (13) is changed to:

$$p = \frac{1}{1+\exp(-\Delta)} \tag{16}$$

Where Δ is the minimum of the difference between new-so and dominating solutions in the archive. New-so is set as current-so with the probability (16). If new-so is non-dominated by all solutions in the archive, it is set as current-so and added to the archive. If new-so dominates some solutions in the archive, it is set as current-so; it is added to the archiveand all dominated solutions are removed from the archive.

5. Factors in Reducing Overfitting in FCM

To reduce overfitting requires having sufficient data vectors in each sub-region of input space. A rule describes the relation between the input(s) and output(s), and each cluster describes a sub-region.

Our first criterion is variance and it states that "to reduce overfitting, each cluster shall have sufficient number of data vectors to minimize the variance of the data vectors in the clusters". A high variance of data vectors in a cluster means that there is dispersion between the observations at that locality which results in a confusing cluster center.

Our second criterion is divergence and it states that "the number of clusters helps to reduce overfitting if the value of divergence is maximized".

The volume of clusters in the model is based on the definition of the mean of clusters.

$$\mu_{clusters} = \frac{1}{K}\sum_{k=1}^{K} v_k \tag{17}$$

In (17), K is the number of clusters and v_k is the cluster centers. Convergence of each cluster can be defined as (18).

$$C(i) = \frac{\sum_{i=1}^{N} u_{j,i}^{m}(x_j - v_i)(x_j - v_i)^T}{\sum_{i=1}^{N} u_{j,i}^{m}} \tag{18}$$

Where, $u_{j,i}{}^{m}$ is defined by (2), N is number of feature vectors and v_i is cluster center of the i^{th} cluster. We define the divergence of the model by

$$D = \frac{\sum_{i=1}^{N}\sum_{j=1}^{K} u_{i,j}{}^{2}}{\sum_{i=1}^{N}\sum_{j=1}^{K} u_{i,j}(v_{i,j} - \mu_{clusters})(v_{i,j} - \mu_{clusters})^T} \tag{19}$$

An appropriate number of clusters, C, and the proper value for fuzzification power, m, can be solved by a multi-objective optimization as (20) according to (18) and (19).

$$\text{optimum } [\, C(N,m), D(N,m) \,] \tag{20}$$

So there are two criteria for proper clusters and these criteria must be noted in the first stage of FCM algorithm. In this paper, two multi-objective optimization algorithms are used to collect these measures for decision making; so the basic algorithm of FCM is modified to choose suitable numbers of clusters and fuzzifier by evolutionary optimization algorithm.

6. Experimental Results

The proposed objective functions for reduction of the overfitting is tested on three data sets which are obtained from [27] by experimental evaluation.

a)A non-linear system with characteristic equation

$y=(1+x_1^{-2}+x_2^{1.5}+x_3+x_4)$;

where ($x_1 \geq 1$, $x_2 \leq 5$); and x_3, x_4 are dummy

b)The measurements from a chemical plant and the setting of a human operator controlling this plant.

c)The daily stock price of a stock a stock market.

In AMOSA, for each data set, a vector of appropriate dimension is considered as a solution for the problem. At first, we need to specify a maximum and minimum number for the cluster centers. Equal to the difference of the minimum and maximum values, cells are considered in the solution and each cell has a binary value. If each cell is one, it means one unit is added to minimum number of centers. For example, if N_{max}=10, N_{min}=2 and cells assigned to number of centers in solution are (1 0 0 1 1 1 0 1). There are 7 centers in model. There is a need to specify the value of fuzzifier (m). Parameter m, is coded by two groups of binary cells, m_{int} and m_{rl} , where m_{int} codes the integer part while m_{rl} codes the fractional part. Two cells are considered to save the decimal value of these numbers, N_{dec} and m_{dec}. Moreover, two cells keep the convergence and divergence, C and D, of model according to (10) and (11). The last cell represents the archive status of solution (in archive or not). So the final format for solution is:

$$[N_{min}, N_{max}, m_{int}, m_{rl}, N_{dec}, m_{dec}, C, D, S] \tag{21}$$

Other parameters of the EOA are:
tmp_{high}=10, tmp_{low}=0, $tmp_{difference}$=0.1, max-iteration =100

Two approaches are used to modify a solution. The first approach sets the binary contents of a random cell randomly such as converting [**0** 1 0 1 1 0 1 0 1] to [**1** 1 0 1 1 0 1 0 1] by randomly determined cell position 1 to modify its contents randomly. The second approach swaps the values of two random cells, i.e., by randomly determined

cell positions 1 and 7 the solution [**1**0 0 0 1 1 **0** 0 1] turns to [**0** 0 0 0 1 1 **1** 0 1] .

The data sets are normalized before the clustering by

$$x_n = (x - x_l) / (x_h - x_l) \quad (22)$$

Where x_h is the highest value of the variable, and x_l is the lowest value of data. Normalization is applied on each input and output independently. The following result is produced for 3 times running this algorithm. (C and D are (18) and (19))

Table 1
Results of AMOSA

Data Set	No. of Clusters	Volume of Fuzz
1	5	15.4
2	4	10.3
3	4	9.8

The optimal number of clusters and the value of fuzzifier which are obtained in the previous stage causes to create 0.05 error in verification data in the first data set, 0.001 error in the second data set, and 0.34 in the third data set.

In MOICA, a 3-dimentions matrix is initiated randomly. In the future steps, the empires are involved in imperialistic competition, and then final results are produced.

Table 2
Results Of Moica

Data Set	No. of Clusters	Volume of Fuzz
1	4	15.3
2	5	9.7
3	3	8.9

7. Conclusion

In this research, two multi-objective optimization algorithms (AMOSA, MOICA) are used for optimization of some formula in FCM algorithm. The hit ratio for trained FCM is compared with hit ratio of FCM with ordinary parameters in the following table.

Table 3
ComparisionBetween Hit Rations

Data Set	Hit Ratio of AMOSA	Hit ratio of MOICA	Hit ratio of ordinary
1	0.91	0.88	0.76
2	0.89	0.93	0.79
3	0.92	0.90	0.74

Although results of using AMOSA and MOICA are analogous, MOICA can reach the final set of solutions in fewer epochs. Instead of these algorithms, using other multi-objective optimization algorithms and evolutionary differential algorithms may be the cause of better result.

References

[1] R. Babuska, H.B. verbuggen, "Constructing Fuzzy Model By Product Space Clustering", Delft University of Technology, P.O. Box 5031, NL_2600 GA Delft, The Netherlands

[2] M. Sugeno, Takahiro Yasukawa, "A Fuzzy-Logic-Based Approach to Qualitative Modeling", IEEE Transaction On Fuzzy Systems, VOL. 1, NO. 1, Feb 1993

[3] J.C. Bezdek, "Fuzzy Mathematics in Pattern Classification", PhD thesis, Cornell University, Aug 1973

[4] L. Wang, "A Course in Fuzzy Systems and Control", Prentice-Hall International Inc.

[5] H. Sun , S. Wang, Q. Jiang, "FCM-Based Model Selection Algorithms for Determining the Number of Clusters", Elsevier, Pattern Recognition 37 (2004) 2027-2037

[6] H. Imai, A. Tanaka, M. Miyakoshi, "A Method of Identifying Influential Data in Fuzzy Clustering", Ieee Transactions On Fuzzy Systems, vol. 6, no. 1, 1998

[7] R. W.Tibshirani, G. Hastie, "Estimating the Number of Clusters in a Dataset via the Gap Statistic", JRSSB (2000)

[8] G. Hamerly, C. Elkan, "Learning the k in k-means", NIPS (2003)

[9] J. Bringer, E. Hullermeier, "Adaptive Optimization of the Number of Clusters in Fuzzy Clustering", IEEE, ISSN 1098-7584, Print ISBN 1-4244-1209-9

[10] C. A. CoelloCoello, D. A. Van Veldhuizen, G. B. Lamont, "Evolutionary Algorithms for Solving Multi-Objective Problems", Speringer, 2nd Ed., 2007

[11] C. Dimopoulos, "A Review of Evolutionary Multi-objective Optimization, Application in the Area of Production Research", IEEE, Vol. 2, DOI:10.1109/CEC.2004.1331072, 2004

[12] C. A. CoelloCoello, "An Updated Survey of Evolutionary Multi-Objective Techniques: State of the art and Future Trends", IEEE, Vol.1, DOI: 10.1109/CEC.1999.781901

[13] Z. Kowalczuk, T.Bialaszewski, "Improving Evolutionary Multi-Objective Optimization Using Genders", Speringer, Vol. 4029/2006, DOI 10.1007/1178231_42

[14] S. Bandyopadhyay, S. Saha, U. Maulik, K. Deb, "A Simulated Annealing-Based Multi-Objective Optimization Algorithm: AMOSA", IEEE, Vol. 12, Issue 3, ISSN 1089-778x, 2008

[15] J. C. Dunn, "A Fuzzy Relative of the ISODATA Process and Its Use In Detecting Compact Well Seperated Clusters", Vol. 3, No. 3, 1974

[16] J. C. Bezdek, "Pattern Recognition With Fuzzy Objective Function Algorithm", New York: Plenum, 1981

[17] L. Gath, A. B. Geva, "Unsupervised Optimal Fuzzy Clustering", Pattern Analysis and Machine Intelligence, IEEE Transactions, vol. 11, Digital Object Identification: 10.1109/34.192473

[18] K.Y. Chan, C.K. Kwong, T.S. Dillon, Y.C. Tsim, "Reducing overfitting in manufacturing process modeling using a backward elimination based genetic programming", ELSEVIER, Applied Soft Computing 11 (2011) 1648-1656

[19] T. Feuring, J.J. Buckley, Y. Hayashi, "Fuzzy Neural Nets Can Solve the Overfitting Problem", IEEE, Digital Object Identifier: 10.1109/IJCNN.1999.830838

[20] H. Nakanishi, I.B. Türkşen, M. Sugeno, A Review and Comparison of Six Reasoning Methods, Fuzzy Sets and Systems 57(3), pp. 257-294, 1993.

[21] F. Chung, T. Lee, "Analytical Resolution and Numerical Identification of Fuzzy Relational Systems", IEEE, Digital Object Identifier: 10.1109/3477.735404

[22] M. Delgado, A.F. Gdmez-Skarmeta, F. Martin, "A methodology to model fuzzy systems using fuzzy clustering in a rapid-prototyping approach", ELSEVIER, Fuzzy Sets and Systems 97 (1998) 287-301

[23] M.Bodur, A.Acan, T. Aykol, "Fuzzy System Modeling with the Genetic and Differential Evolutionary Optimization", IEEE, Digital Object Identifier :10.1109/CIMCA.2005.1631303

[24] A. LotfiZadeh, "Fuzzy Sets", Information and Control, 338-353, 1965

[25] A. LotfiZadeh, "The birth and evolution of fuzzy logic", International Journal of General Systems 17, 95-105, 1990
[26] I. BurhanTurksen, "Fuzzy function with LSE", ELSEVIER, Applied Soft Computing 8 (2008) 1178–1188
[27] H. Nakanishi, I.B. Türkşen, M. Sugeno, A Review and Comparison of Six Reasoning Methods, Fuzzy Sets and Systems 57(3), pp. 257-294, 1993.
[28] A. Khabbazi, E. Atashpaz, C. Lucas, "Imperialist Competitive Algorithm for Minimum Bit Error Rate Beamforming", Inte. J. Bio-Inspired Computation, Vol. 1, 2009
[29] E. Atashpaz, C. Lucas, "Imperialist Competitive Algorithm: An Algorithm for Optimization Inspired by Imperialistic Competition", IEEE, Congress on Evolutionary Computation, 2008
[30] C. Lucas, Z.Nasiri-Gheidari, F. Tootoonchian, "Application of an Imperialist Competitive Algorithm to the Design of a Linear Induction Motor", ELSEVIER, Energy Conversion and Management 51 (2010) 1407-1411
[31] E. Atashpaz, F. Hashemzadeh, R. Rajabioun, C. Lucas, "Colonial Competitive Algorithm: A Novel Approach for PID Controller Design in MIMO Distillation Column Process", Emerald, Int. J. of intelligent computation and cybernetics, VOL.1, NO. 3, 2008.