Zhu, N., Ong, Y.S., Wong, K.W. and Seow, K.T. (2003) Parameter identification using memetic algorithms for fuzzy systems.
In: Proceedings of the Fourth International Conference on Intelligent Technologies (InTech' 03), 17 - 19 December, Chang Mai, Thailand, 833-839.

Parameter Identification Using Memetic Algorithms for Fuzzy Systems

Ning Zhu, Yew Soon Ong, Kok Wai Wong, Kiam Tian Seow

School of Computer Engineering
Nanyang Technological University
Nanyang Avenue, 639798, Singapore

Email: {pg04169798 | asysong | askwwong | asktseow} @ ntu.edu.sg

Abstract: In recent years, fuzzy modelling has become very popular because of its ability to assign meaningful linguistic labels to fuzzy sets in the rule base. However, in order to achieve better performance in fuzzy modelling, parameter identification often needs to be performed. In this paper, we address this optimization problem using memetic algorithms (MAs) for Sugeno and Yasukawa's (SY) qualitative (fuzzy) model. MAs are essentially variants of Genetic Algorithms incorporated with local search methods (or memes) that could better improve the search control accuracy. In addressing the parameter identification problem, MAs are utilized to perform search exploitation within the neighbourhood of the prior knowledge extracted via the Improved SY fuzzy modelling approach. The use of MAs in performing parameter identification is examined empirically, and found to produce better solutions attributable to the extraction and proper use of prior knowledge.

Keywords: fuzzy modelling, genetic algorithms, memetic algorithms, parameter identification.

1. Introduction

Fuzzy modelling has become very popular because of its main feature being the ability to assign meaningful linguistic labels to the fuzzy sets in the rule base [1]. Sugeno and Yasukawa's qualitative modelling (SY) method [2] has gained much attention in the fuzzy research field mainly due to its advantage of building fuzzy rule bases automatically from sample input-out data. The usual fuzzy controller identification methods generate dense fuzzy rule bases, so that the rule premises form a fuzzy partition of the input space. In a dense fuzzy rule base, the number of rules is very high, as it depends on the number of inputs and the number of partitions per variable in an exponential way. In order to avoid this exponential number of rules, the SY method puts emphasis on the rule consequents, i.e., the output space, and first finds a partition in output space. The determination of premises in the input space is done by splitting appropriately the inverse images of the output clusters.

However, in the original SY modelling, there are a few issues left unaddressed. This has lead to an improvement to the original SY method known as the Improved SY modelling [3]. However, the prediction or control accuracy of such fuzzy model depends largely on how well the parameters are identified i.e. parameters identification, and that is the focus of this paper. Previous work has been done in this area [2,3] to optimize parameters. In this paper, we propose using memetic algorithms to improve this process, which is an optimizing process.

In the area of optimization, there are two kinds of search methods: global search and local search. Global optimizers are useful when the search space is likely to have many minima, making it hard to locate the true global minimum [4]. Examples of global search methods are genetic algorithms. Genetic algorithms are search algorithms that use operators found in natural genetics to guide the trek through a search space. They are empirically proven to provide robust search capabilities in complex spaces, but they usually take a relatively long time to convergence to the exact optimum, and in the worst case, it may never converge.

On the other hand, local improvement procedures can quickly find the exact local optimum of a small region of the search space, but are typically poor global searchers. Because local procedures do not guarantee optimality, in practice, several random starting points may be generated and used as input into the local search technique and the best solution is recorded. This optimization technique, commonly known as multi-start algorithm has been used extensively. Nevertheless it is a blind search technique since it does not take into account past information [5]. Genetic algorithms, unlike multi-start, utilize past information in the search process. Therefore, local improvement procedures have been incorporated

into GAs to improve their performance through what could be termed "learning". Such hybrid GAs, which utilizes local learning heavily, are known as memetic algorithms. These techniques have been used successfully to solve a wide variety of realistic problems and will be used here in this paper [6].

There are two competing goals governing the design of the global search methods: exploration is to ensure global reliability; exploitation concentrates the search efforts around the best solutions found so far by searching their neighborhoods to produce better solutions [7].

In this paper, our key focus is to investigate the use of memetic algorithms for parameter identification in fuzzy system. In our design of the memetic algorithms, we use the strategy of exploitation. With the help of prior knowledge in the format of fuzzy rules, precise search range for each parameter can be defined. It is believed that memetic algorithm is able to improve on the parameter identification process, and thus improving the accuracy of the fuzzy model. This paper is organized as follows: Section 2 briefly describes the SY fuzzy modeling. Section 3 presents the parameter identification problem in SY fuzzy modelling and related works in the area. Section 4 describes the canonical memetic algorithms. Section 5 presents the fuzzy modelling using memetic algorithms for parameter identification. The empirical results using synthetic data are presented in section 6 while section 7 summarizes the main conclusions.

2. SY Fuzzy Modelling

The goal of the SY fuzzy modelling method is to create a transparent, viz. linguistic interpretable fuzzy rule based model from input-output sample data [2]. The construction of the rule base is performed in two main steps: the identification and the build-up of the qualitative model. The former can be further divided into two tasks: the structure identification and parameter identification. Having an identification model at hand, linguistic labels can be assigned to the finalized fuzzy sets in the rules in the qualitative modelling phrase. Structure identification may be classified into two categories [2]. The identification task is summarized in Table 1.

Structure identification I	a: input candidates
	b: input variables
Structure identification II	a: number of rules
	b: partition of the input space
Parameter identification	

Table 1: Classification of the identification

Type I structure identification consists of finding the input candidates of the system and the variables that affect the output significantly. Type II structure identification covers the determination of the number of rules and the partition of the (usually) multidimensional input space.

Assume the given data set; $x_1, \ldots x_n$ are the input variables, y is the output variable. N sample data are given in the form of $\left(x_1^i, x_2^i, ..., x_n^i\right) \rightarrow y^i, i = 1, ..., N.$

2.1. Identification of input candidates and variables

The structure identification type Ia consists of finding possible input candidates for the input to a system. There are an infinite number of possible candidates, which should be restricted to a certain number. In general, the selection of the input candidates is not a systematic process, i.e., one has to take a heuristic method based on experience and/or common sense knowledge for this purpose.

The structure identification of type Ib concerns with the selection of input variables that truly influence the output. In other words, one has to choose a set of effective variables among a finite set of original variables. For this purpose one needs a criterion function to evaluate the various candidate sets of variables. This function assigns a value for a given set of variables and its task is to minimize or maximize it. In [2], the regularity criterion (RC) [8] is employed, which was performed between steps identification of type II and parameter identification. The outcome of the RC method depends on the identification of type II. The RC is a heuristic method that selects a set of inputs among the possible candidates.

In the first step, the sample data set is divided into two groups, A and B. The criterion function is

$$RC = \left[\sum_{i=1}^{K_A} \frac{\left(y_i^A - y_i^{AB}\right)^2}{K_A} + \sum_{i=1}^{K_B} \frac{\left(y_i^B - y_i^{BA}\right)^2}{K_B} \right] \Big/ 2 \quad (1)$$

where k_A and k_g are the numbers of data in groups A and B, respectively; y_i^A and y_i^B are the outputs of groups A and B, respectively; and finally y_i^{AB} (y_i^{BA}) is the model output for the group A (B) input estimated by the model identified using group B (A) data. For evaluating equation (1) two models should be built from the data groups A and B at each evaluation stage.

2.2. Determination of the number of rules and the input partition

Usually in the design of a fuzzy system the rule antecedents and the partition of the input domain are determined. This (dense) rule base design methodology results in exponentiality in terms of the number of rules.

To avoid this significant drawback, the SY method proceeds inversely: first the partition of the output space is determined, which is done by clustering the whole output data set using fuzzy c-means clustering (FCMC)[9]. The optimal numbers of cluster are determined by means of the following criterion [10]:

$$SC = \sum_{k=1}^{N}\sum_{i=1}^{C}(\mu_{ik})^m\left(\|x_k - v_i\|^2 - \left\|v_i - \bar{x}\right\|^2\right) \quad (2)$$

where N is the number of data to be clustered; C is the number of clusters, $C \geq 2$; x_k is the k^{th} datum (usually vector), $\bar{x}$ is the mean data x_k; v_i is the center of the i^{th} cluster (vector); μ_{ik} is the membership degree of the k^{th} datum with respect to the i^{th} cluster; m is the fuzzy exponent, m > 1.

As a result of the clustering, every output datum is associated with a membership degree in all the clusters B_i, where i = l,..., C. From an output fuzzy clusters B_i we can induce a fuzzy cluster A_i in the multi-dimensional input space. This cluster can be projected onto the axes of the variables, hence defining the antecedent fuzzy sets in each input dimension. Starting from a cluster B_i, and assuming that we have two input variables x_1 and x_2, we usually obtain a rule like

If x_1 is A_{i1} and x_2 is A_{i2} then y is B_i.

Although this notation implies that the number of rules is identical with the number of output clusters, it can happen that this is not the case [3].

2.3. Parameter modelling

Parameter modelling step can be accomplished in two stages in the fuzzy model design. The authors in [2] proposed to repeat it in every input candidate evaluation step. At this stage we have to measure the performance of the rough fuzzy model. For this purpose the following performance index (PI) is used:

$$PI = \sum_{i=1}^{m}\left(y^i - \hat{y}^i\right)^2 \Big/ m \quad (3)$$

where m is the number of the data, y^i is the i^{th} actual output, and $\hat{y}^i$ is the i^{th} model output.

In the case of fuzzy model, the parameters are those of the membership functions. Utilizing trapezoidal membership functions implies there are 4 parameters to be optimized for each antecedent $p_1 \leq p_2 \leq p_3 \leq p_4$.

3. Parameter identification

In this paper, our main focus is on parameter identification in SY fuzzy modelling. The following outlines a previous algorithm [2] used to identify the parameters.

1) Set the value *f* of adjustment.
2) Assume that the k^{th} parameter of the j^{th} fuzzy set is p_j^k.
3) Calculate $p_j^k + f$ and $p_j^k - f$. If k=2,3,4,and $p_j^k - f$ is smaller than p_j^{k-1}, then $\hat{p}_j^k = p_j^{k-1}$; else $\hat{p}_j^k = p_j^k - f$. also if k=1,2,3 and $p_j^k + f$ is bigger than p_j^{k+1}, then $\hat{p}_j^k = p_j^{k+1}$; else $\hat{p}_j^k = p_j^k + f$. *f* is a constraint for adjusting parameters.
4) Choose the parameter which shows the best performance PI in among $\left\{\hat{p}_j^k, p_j^k, \hat{\hat{p}}_j^k\right\}$ and replace p_j^k with it.
5) Go to step 2 while unadjusted parameter exist.
6) Repeat step 2 until we are satisfied with the result.

In [2], 5% of the width of the universe of discourse is used as the value of *f* and repeat steps 1 to 6 over 20 iterations.

The accuracy of the SY fuzzy modeling was further improved in [3], by modifying the original parameter identification procedure, which works with temporally changing adjusting value *f* depending on the actual performance value. They set the starting adjusting value in the k^{th} input as

$$f = dom(k) \Big/ 4p_{steps} \bullet 2^{p_{start}+m} \quad (4)$$

where dom (k) is the domain of the k^{th} input, i.e., the difference between the smallest and the largest input in the given dimension; p_{steps} a predefined constant (default: 3); p_{start} is to set the starting precision (default: -1), and m is an iteration counter which increases if

$$PI_{speed} = \frac{PI_{last} - PI_{actual}}{PI_{last}} < 0.1 \quad (5)$$

that is, if the amelioration of the performance index is less than 10%. The starting g value is zero. The parameter identification is organized in a double loop. In the inner loop the four parameters of the trapeze membership function of all the antecedents are sequentially adjusted with the same actual adjusting value until no improvement can be achieved or the number of inner iteration attains a certain limit. Then m is increased if the equation (5) holds, and the whole process restart again. The

stopping criterion of the outer loop can be either the crossing of a certain time limit of when the PI_{speed} gets smaller than a certain threshold. However, the realization of the optimal set of fuzzy parameters depends very much on the assumption made to some constant parameters in equation (4) and (5). This has led us in search for an optimizing algorithm to further improve this parameter identification process.

4. Memetic Algorithms

Memetic Algorithms are population-based approaches for heuristic search in optimization problems [11]. Basically, they are genetic algorithms that apply a separate local search process to refine individuals. One big difference between memes and genes is that memes are processed and possibly improved by the people that hold them - something that cannot happen to genes. Experimental results show that the memetic algorithms have better results over simple genetic or evolutionary algorithms [12, 13].

4.1. Genetic Algorithms

Genetic algorithm based on the Darwinian survival-of-the-fittest theory, is an efficient and broadly applicable global optimization algorithm [14]. In contrast to conventional search techniques, genetic algorithm starts from a group of points coded as finite length alphabet strings instead of one real parameter set. Furthermore, genetic algorithm is not a hill-climbing algorithm hence the derivative information and step size calculation are not required. The three basic operators of genetic algorithms are: selection, crossover and mutation. It selects some individuals with stronger adaptability from population according to the fitness, and then decides the copy number of individual according to the selection methods such as Backer's stochastic universal sampling. It exchanges and recombines a pair of chromosome through crossover. Mutation is done to change certain point state via probability. In general, the range of crossover probability is 0.5~ 1.0 and the range of mutation probability is 0.005~0.1, one needs to choose suitable crossover and mutation probability time and again via real problems.

4.2. Local Search Methods

Local search method is a method of searching a small area around a solution and adopting a better solution if found. Typical local search method is outlined in figure 1. The search begins with choosing a direction of movement is prescribed according to some algorithm, and a line search or trust region approach is performed to determine an appropriate next step. The process is repeated at the new point and the algorithm continues until a local minimum is found. Schematically, a model local minimizer method can be sketched as follows:

Procedure A Typical Local Search Method
Begin

1. Start from some given point x_1
2. Assign count k=1;
3. Calculate a search direction D_k
   Determine an appropriate step length λk
   Replace $x_{k+1} = x_k + \lambda_k * D_k$
4. Converged?
   - Yes: stop and output results
   - No: goto step 3.

End

Figure 1: A typical Local Search Method

Different methods are distinguished by their choice of search direction. In general according to algorithms that use derivative or not, continuous parametric local search methods can be categorized as second, first or zeroth order method [15]. In our paper, we have make use of all of them.

4.2.1. Second order method

This method requires the function values, its first (partial) derivative vector and the second derivative matrix - the Hessian. Newton-Raphson method is an example of this kind. It is based on the idea of approximating partial derivative with its linear Taylor series expansion about a value. This method is powerful and simple to implement. It will converge to a fixed point from any sufficiently close starting value.

4.2.2. First order method

When the second derivative is not available, we use only function values and first (partial) derivative vector. It includes Steepest Descent, Conjugate Gradients and Quasi-Newtonian Methods. Fox example, steeping descent uses the first-order Taylor polynomial to construct local linear approximation of the function.

4.2.3. Zeroth order method

Zeroth order methods, also known as direct search techniques, are often very useful when the derivative information, both Hessians and gradients of the function, are either unavailable or unreliable. The main feature of zeroth order method is that they only need the object function values, or even only need the relative rank of objective values. The direct search only needs this information to guarantee the sufficient information about the local behavior of the function. Pattern search, simplex search and methods with adaptive sets of search directions constitute the three main categories of direct search.

4.3. Pseudo- code for Canonical Memetic Algorithms:

In the first step, the GA population is initialized randomly. Subsequently, for each chromosome in the population, a local search is used for local improvement based on the Lamarckian or Baldwin evolution. Standard GA operators are then used to generate the next population until the stopping conditions are satisfied.

Procedure Canonical Memetic Algorithms
Begin
Initialize: Generate an initial GA population;
While (Stopping conditions are not satisfied)
Evaluate all individuals in the population
For each individual in the population
- Perform local search on it
- Replace it with locally improved solution

End For
Apply standard genetic algorithm operators to create a new population.
End while
End

Figure 2: The pseudo-code for Canonical memetic algorithms

5. Parameter Identification using Memetic Algorithms

Here our main purpose is to use memetic algorithms to exploit the promising region. Hence we concentrate our search efforts on the best solutions so far by searching their direct neighbourhood to produce better solutions.

The Improved SY modelling [3] is used as the starting points to provide the prior knowledge, then we search the nearest region of each parameter as below:

$$low-bound \le x1 \le \left(\hat{p}1+\hat{p}2\right)\Big/2$$
$$\left(\hat{p}1+\hat{p}2\right)\Big/2 \le x2 \le \left(\hat{p}2+\hat{p}3\right)\Big/2$$
$$\left(\hat{p}2+\hat{p}3\right)\Big/2 \le x3 \le \left(\hat{p}3+\hat{p}4\right)\Big/2$$
$$\left(\hat{p}3+\hat{p}4\right)\Big/2 \le x4 \le up-bound \qquad (6)$$

where $\hat{p}1$, $\hat{p}2$, $\hat{p}3$, $\hat{p}4$ are the fuzzy parameters generated after the parameters identification process from the improved SY fuzzy model

---

Procedure Parameter identification using memetic algorithms
BEGIN
Step 1: Initialize
- Generate an initial MA population according to each parameter's own range as in (6).

Step 2: **For** each individual in the population
- Perform local search on it according to corresponding range.
- Replace the genotype in the population with the locally improved solution.

End For
Step 3:
- Evaluate all individuals in the population.
- Apply standard MA operators to create a new population; i.e., Selection, Mutation and Crossover.

Step 4: Stopping condition satisfied?
- Yes: stop and output results
- No: goto step 2.

END

Figure 3: Parameter identification using memetic algorithms

6. Case Study and Discussion

In this section, we present the optimizing performance of the proposed parameter identification technique. As memetic algorithm uses different local search algorithms to refine individuals in the optimizing process, comparison results will be presented in this section. Synthetical sample data [2] from fuzzy systems is used here. The memes or local search methods employed also consists of many derivative and non-derivative methods.

6.1. Test data

Here we consider the following nonlinear static system with two inputs: x_1 and x_2, and a single output y:

$$y = \left(1 + x_1^{-2} + x_2^{-1.5}\right)^2, \quad 1 \le x_1, x_2 \le 5 \qquad (7)$$

From this system, 50 input-output data are obtained (table 2).

Group A				Group B			
No.	X_1	x_2	y	No.	x_1	x_2	y
1	1.40	1.80	3.70	26	2.00	2.06	2.52
2	4.28	4.96	1.31	27	2.71	4.13	1.58
3	1.18	4.29	3.35	28	1.78	1.11	4.71
4	1.96	1.90	2.70	29	3.61	2.27	1.87
5	1.85	1.43	3.52	30	2.24	3.74	1.79
6	3.66	1.60	2.46	31	1.81	3.18	2.20
7	3.64	2.41	1.95	32	4.85	4.66	1.30
8	4.51	1.52	2.51	33	3.41	3.88	1.48
9	3.77	1.45	2.70	34	1.38	2.55	3.14
10	4.84	4.32	1.33	35	2.46	2.12	2.22
11	1.05	2.55	4.63	36	2.66	4.42	1.56
12	4.51	1.37	2.80	37	4.44	4.71	1.32
13	1.84	4.43	1.97	38	3.11	1.06	4.08
14	1.67	2.81	2.47	39	4.47	3.66	1.42
15	2.03	1.88	2.66	40	1.35	1.76	3.91
16	3.62	1.95	2.08	41	1.24	1.41	5.05
17	1.67	2.23	2.75	42	2.81	1.35	1.97
18	3.38	3.70	1.51	43	1.92	4.25	1.92
19	2.83	1.77	2.40	44	4.61	2.68	1.63
20	1.48	4.44	2.44	45	3.04	4.97	1.44
21	3.37	2.13	1.99	46	4.82	3.80	1.39
22	2.84	1.24	3.42	47	2.58	1.97	2.29
23	1.19	1.53	4.99	48	4.14	4.76	1.33
24	4.41	1.71	2.27	49	4.35	3.90	1.40
25	1.65	1.38	3.94	50	2.22	1.35	3.39

Table2: Input-Output data of nonlinear system

6.2. Local Search Methods

Four local search methods are employed in this empirical study. They consist of a variety of optimization methods from the Schwefel libraries [16] and a few others in the literature. They are representations of second, first and zeroth order methods.

6.2.1. DP

Schwefel library [16] DSCP searches that uses the strategy of Davis, Swann and Compey with Palmer orthogonalization [17]. Small changes in all variables are tested one at a time to establish a suitable direction for a simple constant direction search, followed by an orthogonalization to establish a new set of rotated co-ordinates in which to carry out the small step tests.

6.2.2. DC

A dynamic hill climbing algorithm is available [18]. This method uses a series of gradient descent local searches to find individual optimal solutions, and then they are used to find new starting points. It aims to locate as many individual optimal solutions as possible with each one being found by a rapid local search.

6.2.3. HO

Hooke and Jeeves Direct Search [19]. This kind of method is characterized by a series of exploratory moves that consider the behavior of the objective function at a pattern of points, all of which lie on a rational lattice. The exploratory moves consist of a systematic strategy for visiting the points in the immediate vicinity of the current iterate.

6.2.4. RN

Rosenbrock's rotating co-ordinate search [20]. This method in a similar manner to the Hooke and Jeeves search bet additionally allows the directions of search to alter so that it is not restricted to the co-ordinate system based on the individual variable directions.

6.3. Empirical Study

The propose method use memetic algorithms for improving the parameter identification; that is, we use the SY modelling to extract the prior knowledge, follow by using memetic algorithms to perform search for fine tuning.

In the empirical study, we employ a standard binary coded genetic algorithm for the memetic search. Backer's stochastic universal sampling algorithm is used here. Mutation is done via randomly selecting a bit and flipping it. In our test, we allows it to continue until the maximum of 100,000 trials, and the control parameters for memetic algorithms were set using default values as follows: population of 50, mutation rate of 0.1%, uniform crossover with a rate of 60%, 10 bit binary encoding, and maximum local search length of 500 evaluations. We use the PI in equation (3) as the fitness function. Experiment results are as below:

MAs with different loca search	From Improved SY fuzzy model	After MA fine tuning
GA-DP	0.044115	0.020864
GA-DC	0.044115	0.015432
GA-HO	0.044115	0.024392
GA-RN	0.044115	0.017787

Table 3: Comparison of PI before and after optimization

The results of those before optimization are starting points of our search, then we search their direct neighbourhood using memetic algorithms, and we get the results as shown in Table 3 after optimization.

From the results, we can see that memetic algorithms can achieve better results overall, by comparing to the results generated from the Improved SY modelling [3]. So we can conclude that the design of exploitation by an intensified search in a promising region of the search space can be successfully applied to parameter identification of the fuzzy systems.

Next, it is shown that GA-DC achieve best results, this is because it works very well for multi-

modal problems with a moderate number of peaks and relatively smooth surfaces.

7. Conclusion

In this paper, we propose a technique to enhance the parameter identification of the SY modelling using memetic algorithm. In essence it is the idea of exploiting the direct neighbour of the best solutions found do far to produce better solutions. The set of fuzzy rules generated by the Improved SY fuzzy modelling acts as a prior knowledge for the memetic algorithm. Memetic algorithm, a hybrid genetic algorithm-local search methods, which incorporate local improvement procedures with traditional GAs may thus be used to improve the performance of GAs in search, and thus further improve the performance of the parameter identification.

8. References

[1] L.A. Zadeh, *Fuzzy Algorithm*, Information and Control, vol (12), Page 94 - 102, 1968.

[2] M. Sugeno, and T. Yasukawa, *A Fuzzy Logic Based Approach to Qualitative Modelling*, IEEE Transactions on Fuzzy System, volume 1 no. 1, page 7-31,1993.

[3] D. Tikk, G. Bíró, L.T. Kóczy, T.D. Gedeon, K.W. Wong, *Notes on Sugeno and Yasukawa's fuzzy modelling approach*, Proceedings of Joint 9th IFSA World Congress and 20th NAFIPS International Conference, Vancouver, Canada, page 2836 – 2841, July, 2001.

[4] D.E.Goldberg, Genetic algorithms in search, optimization and machine learning, Addison-Wesley, 1989

[5] A. Rinnooy Kay and G. Timmer, *Stochastic global optimization methods part i*, Mathematical Programming, 39(1), page 27-56,1987

[6] P. Moscato and M.G. Norman, *A memetic approach for the traveling salesman problem – implementation of a computational ecology for combinatorial optimization on message-passing systems*, in proceeding of the International Conference on Parallel Computing and Transputer Applications. IOS Press (Amsterdam) 1992.

[7] A.Torn and A.Zilinskas , *global optimization,*Volume 350 of Lecture Notes in Computer Science. Springer-Verlag, 1989.

[8] J. Ihara. *Group method of data handling towards a modelling of complex system IV*. Systems and Control, 24, page 158-168, 1980.

[9] J.C.Bezdek. *Pattern Recognition with Fuzzy Objective Function Algorithms.* Plenum Press, New York, 1981.

[10] Y.Fukuyama and M.Sugeno. *A New method of choosing the number of clusters for fuzzy c-mean method.* In proc. Of the 5th Fuzzy System Symposium, page 247-250, 1989.

[11] P.Moscato *On evolution, search, optimization genetic algorithms and martial arts: towards memetic algorithms,* Caltech Concurrent Computation Program, C3P Report 826, 1989.

[12] W.E.Hart, *adaptive global optimization with local search,* Phd thesis, university of Califonia, San Diego, May 1994.

[13] P..Merz, *on the performance of memetic algorithms in combinatorial optimization, 2th* workshop on memetic algorithms, GECCO, San Francisco, CA, USA,2001.

[14] J. Holland, *Adaptation in Nature and Artificial Systems*, Ann Arbor, MI: University of Michigan Press 1975

[15] B. D. Bunday, *Basic Optimization Methods*, Edward Arnold (Publishers) Ltd, 1985.

[16] H. P. Schwefel, *Evolution and Optimum Seeking*, John Wiley &Son, 1995.

[17] W. H. Swann, *A report on the development of a new direct searching method of optimization*, ICI CENTER Instrument Laboratory, Middlesborough, 1964.

[18]D.Yuret and M.de la Maza, *Dynamic Hill Climbing: Overcoming the Limitation of Optimization Techniques*, page 254-260 in Proceedings of the 2nd Turkish Symposium on AI and ANN, 1993.

[19]J. L. Horner, R. G. White, R. Hooke and T. A. Jeeves*, Direct search solution of numerical and statistical problems*, J. Sound Volume 147(1) page 87-103, Westinghouse Research Labs. Scientific Paper, 1960.

[20] H.H.Rosenbrock, An automatic method for finding the greatest or least value of a function, Comp.J. Volume 8 page 42-52, 1965.