Conférence Internationale en Automatique & Traitement de Signal (ATS-2018)
Proceedings of Engineering and Technology – PET
Vol.35pp.95-100

Bat Algorithm Optimization for Fuzzy Rule Base Design of a Fuzzy Controller

Nesrine TALBI
Department of electronic, LAJ Research Laboratory
Faculty of sciences and Technology, Jijel University
Jijel, ALGERIA
t_nesrine2003@yahoo.fr

Abstract— **This paper proposes a new approach to designing fuzzy rule base for a fuzzy controllers using Bat optimization algorithm. The Bat algorithm is one of the most recent swarm intelligence based algorithms that simulates the intelligent hunting behavior of the bats found in nature. The main objective is to optimize the fuzzy rule base of fuzzy controller for dynamic, complex and highly nonlinear systems, respecting the desired performance. To demonstrate the efficiency of the suggested approach, a control of a water bath temperature is selected. Simulation results showed that the proposed approach could be employed as a simple and effective optimization method for achieving optimum determination of fuzzy rule base parameters.**

Keywords- Fuzzy rule base, Bat algorithm, Fuzzy controller, water bath temperature.

I. INTRODUCTION

Fuzzy logic excels in imprecise knowledge representation or incomplete [1]. It provides a convenient interface for modeling of natural language, especially linguistic concepts used by experts of a process. It helps manage complex systems in a simple and easily explainable by human expertise. The difficulty of implementation lies in the development of parameters and membership functions. The choice of the membership functions, their number, the defuzzification, or even of the fuzzy inference, is usually arbitrary.

The Design of a fuzzy rule base is the process that led to the formalization in the form of rules and/or learned relations, from a set of examples between the inputs and outputs of a process. In many cases, the structure is determined empirically by choosing a priori the type of relational approximate reasoning, the number of fuzzy sets for each input variable, and taking all possible combinations to build the fuzzy rule base. So, it is important to mention the difficulty of ensuring consistency & interpretability of fuzzy rules, in particular for multivariable systems where the number of rules becomes very high [2].

The adjustment by successive trial of FS parameters is quite long and tedious. Various techniques of optimization and learning fuzzy systems have been developed. First, Neural networks are used for setting parameters of the premises membership functions in [3] and fuzzy rule consequences in [3]-[5]. After, Gradient descent is used for tuning parameters of the premises and consequences of a FS [6]. After, many improvements have been made using hybridization of fuzzy logic and neural networks [7] - [9].

Another technique of tuning fuzzy systems was the use of metaheuristics, including evolutionary algorithms [10]-[12], and swarm intelligence [13]-[16]. The vast majority of heuristic and metaheuristic algorithms have been derived from the behavior of biological systems and/or physical systems in nature [17]. For example, particle swarm optimization is inspired by social behavior of animals moving in swarm [18], while simulated annealing is based on the annealing process of metals [19]. New algorithms are also emerging recently, including harmony search [20], the firefly algorithm [21], and the Bat algorithm (BA) [17], [22]. The BA technique is introduced by Yang in 2010. It is based on the echolocation behavior of bats. The capability of echolocation of micro-bats is fascinating as these bats can find their prey and discriminate different types of insects even in complete darkness [22]. Bat algorithm was successfully applied to a number of very different problems [23]-[26]. BA is simple to implement and produces good results [26].

This paper presents a simple and effective method using Bat algorithm for designing the fuzzy rule base of a fuzzy controller of type Takagi-Sugeno zero order. Two control problems are considered to study the performance of the proposed algorithm. The remaining of this paper is organized as follows: The structure of fuzzy controller to be optimized is described in the next section. After, the standard BA algorithm is briefly presented in the section III. Section IV explained the method of designing the fuzzy rule base by BA. The test of the effectiveness of the proposed method is made in section V. Finally, the paper is ended by a conclusion.

II. FUZZY CONTROLLER STRUCTURE

This section describes the fuzzy controller (FC) to be designed in this study. The FC is of type Takagi–Sugeno zero order. The i^{th} rule, which is denoted as R_i , is represented in the following form:

$$R_i: If\ e(k) is\ A_{i1}\ and\ \Delta e(k)\ is\ A_{in}$$

ISSN 2356-5608

Conférence Internationale en Automatique & Traitement de Signal (ATS-2018)
Proceedings of Engineering and Technology – PET
Vol.35pp.95-100

$$Then\ u(k)\ is\ o_i \quad (1)$$

where k is the time step, $e(k), \Delta e(k)$ are the input variables, $u(k)$ is the fuzzy controller output variable, A_{ij} is a fuzzy set, and o_i is a crisp value. A_{ij} is a fuzzy set which uses a triangular membership function defined by following equation:

$$\mu(x) = \max(min\left(\frac{x-a}{b-a}, \frac{c-x}{c-b}\right), 0) \quad (2)$$

where a, b and c represent respectively the locations of starting point, peak point and the ending point, for a triangle shaped membership function.

About the fuzzy rule base, the decision on the number of fuzzy rules is a very important issue because it plays a very important role in fuzzy control systems. Unfortunately, there is no systematic and effective procedure for selecting the number of the most appropriate rules, except for some proposals. A reasonable number of fuzzy rules, without losing too much information about the system to be controlled must be carefully obtained.

For flexibility of the implementation of fuzzy controller, the universe of discourse of inputs and output is limited to a range of [-1, 1], determined by the normalization of inputs and output [27]. To do this, the scale factors are used to have the desired dynamics.

This paper proposes a fuzzy rule base composed of only three enabled rules, extracted from analysis expressed as follows:

$\boldsymbol{R_1}: If\ e(k)\ is\ N(-1, a_1, a_2)\ and\ \Delta e(k)\ is\ N\ (-1, b_1, b_2)$

$then\ u(k) = o_1.$

$\boldsymbol{R_2}: If\ e(k)\ is\ Z(a_1, a_2, a_3)\ and\ \Delta e(k)\ is\ Z(b_1, b_2, b_3)$

$then\ u(k) = o_2.$

$\boldsymbol{R_3}: If\ e(k)\ is\ P(a_2, a_3, 1)\ and\ \Delta e(k)\ is\ P(b_2, b_3, 1)$

$then\ u(k) = o_3.$

where $e(k)$ is the difference between the desired output and the measured output of the controlled system. N, Z and P are fuzzy sets of input variables, and o_1, o_2, o_3 are real values of fuzzy controller output.

In the inference mechanism, the AND in the fuzzy rule is implemented by the algebraic product in the theory of fuzzy logic (according to Larsen) .Thus, given a set of input data $\vec{x} = (e, \Delta e)$, the degree of activation $\gamma_i(\vec{x})$ of Rule i is calculated by:

$$\gamma_i(\vec{x}) = \mu_{A_i}(e(k)) \cdot \mu_{B_i}(\Delta e(k)) \quad (3)$$

If there are n_r rules in fuzzy controller, the resulting output of the set of rules is given by the average of weighted individual outputs as follows:

$$u = \frac{\sum_{i=1}^{n_r} \gamma_i(\vec{x}).\, o_i}{\sum_{i=1}^{n_r} \gamma_i(\vec{x})} \quad (4)$$

where o_i is the value of the conclusion of the i^{th} rule.

In this paper, the optimization of the fuzzy controller includes the determination of all parameters of each fuzzy rule.

III. BAT ALGORITHM

Bat algorithm is a metaheuristic optimization algorithm developed by Xin-She Yang in 2010 [22]. The algorithm is based on echolocation of micro-bats with varying pulse rates of emission and loudness. Bats use sonar echoes to detect and avoid obstacles: they determine the size of an object, how far away they are, how fast they are travelling and even their texture, all in split in a second. After hitting and reflecting, bats transform their own pulse to useful information to gauge how far away the prey is. Bats are using wavelengths, that vary from range [0.7, 17] mm or inbound frequencies [20,500] kHz. By implementation, pulse frequency and rate has to be defined. Pulse rate can be simply determined from range 0 to 1, where 0 means there is no emission and by 1, bats are emitting maximum [28], [29], [30].

In order to transform these behaviors of bats to algorithm, Yang used three generalized rules [22]:

1) All bats use echolocation to sense distance, and they also guess the difference between food/prey and background barriers in some magical way.
2) Bats fly randomly with velocity v_i at position x_i with a fixed frequency f_{min}, varying wavelength λ and loudness A_0 to search for prey. They can automatically adjust the wavelength of their emitted pulses and adjust the rate of pulse emission r from [0, 1], depending on the proximity of their target.
3) Although the loudness can vary in many ways, we assume that the loudness varies from a positive large value A_0 to a minimum constant value A_{min}.

The initial position x_i, velocity v_i and frequency f_i are initialized for each bat. For each time step t , the movement of the virtual bats is given by updating their velocity and position using the following equations:

$$f_i = f_{min} + (f_{max} - f_{min}).\, \rho \quad (5)$$

$$v_i^j(t) = v_i^j(t-1) + \left[x_{gbest}^j - x_i^j(t-1)\right].\, f_i \quad (6)$$

$$x_i^j(t) = x_i^j(t-1) + v_i^j(t) \quad (7)$$

where $\rho \in [0,1]$ indicates randomly generated number, and x_{gbest}^j represents current global best solutions.

For the local search part, once a solution is selected among the current best solutions, a new solution for each bat is generated locally using random walk:

ISSN 2356-5608

$$x_i^{new} = x_i^{old} + \sigma . A_{mean}^{old} \quad (8)$$

Where, $\sigma \in [0,1]$ is a random number and represents direction and intensity of random-walk. A_{mean}^{old} is the average loudness of all the bats.

Based on these approximations and idealization, the basic steps of the Bat Algorithm (BA) can be summarized as the following pseudo code [17]:

```
Objective function f(x), x = (x_1, ..., x_d)^T
Initialize the bat population x_i and v_i for i = 1, n
Define pulse frequency Q_i ∈ [Q_min, Q_max]
Initialize pulse rates r_i and the loudness A_i
while (t < T_max) // number of iterations
   Generate new solutions by adjusting frequency, and
   updating velocities and locations/solutions
   if (rand(0,1) > r_i )
         Select a solution among the best solutions
         Generate a local solution around the best
solution
   end if
   Generate a new solution by flying randomly
    If (rand(0,1) < A_i and f(x_i) < f(x))
    Accept the new solutions
   Increase r_i and reduce A_i
    end if
    Rank the bats and find the current best
end while
Post process results and visualization
```

IV. Designing Fuzzy Rule Base of Fuzzy Controller using Bat Algorithm

In this section, we propose to use Bat algorithm for designing a reduced fuzzy rule base of a fuzzy controller of type Takagi-Sugeno zero order, in order to obtain better performances of the system to control.

A. Optimization Procedure

We can summarize the optimization procedure of the fuzzy controller using Bat algorithm through the following steps:

1. Generation of an initial population of solutions characterizing the controller settings.
2. For all solutions:
   - Evaluate the objective function.
   - Classify obtained solutions according to their fitness.
   - Construction of a new population by updating process of frequencies, velocities and solutions.

The step 2 is repeated until a maximum number of iterations is performed. After the process of evolution, the final iteration of the algorithm consists of the well-adjusted solution who provide best solution.

The inclusion of design constraints in the optimization process helps to preserve the semantics of fuzzy rules. For that, the constraints on the limits of the parameter vector to be identified, and limits on the control variables are imposed.

B. Parameters's Vector Representation of Fuzzy Controller

The parameter vector (solution) x of the fuzzy controller has nine parameters. These parameters represent the starting point locations, the pic point, and end point, for a triangular membership function belonging to the inputs of a fuzzy controller and fuzzy singleton for its release.

So $$x = [\, a_1\, a_2\, a_3\, b_1\, b_2\, b_3\, o_1\, o_2\, o_2\, o_3 \,] \quad (9)$$

While respecting the following constraint:

$$\begin{cases} a_1 < a_2 < a_3 \\ b_1 < b_2 < b_3 \\ o_1 < o_2 < o_3 \end{cases} \quad (10)$$

C. Generation of Initial Population of solutions

Initial population is randomly generated from real-valued vectors with dimension d and number of bats n, by taking into account lower and upper boundaries. In this study, $d = 9$.

$$x_i^j = x_{min}^j + rand(0,1) * (x_{max}^j - x_{min}^j) \quad (11)$$

where $i = 1, n; j = 1, d$ and x_{min}^j and x_{max}^j are lower and upper boundaries for dimension j respectively.

V. Application

The test of effectiveness of the Bat algorithm is illustrated to generate fuzzy rule base of fuzzy controller of type Takagi-Sugeno zero order for control of water bath temperature.

A. Water bath temperature control system

The objective of this section is to control the temperature of a water bath. The structure of the process is given by the diagram of the Fig. 1.

The goal of this simulation is to control the temperature of a water bath system given by the following equation [31]:

$$\frac{dy(t)}{dt} = \frac{u(t)}{C} + \frac{Y_0 - y(t)}{RC} \quad (12)$$

where $y(t)$ is the system output temperature in °C; $u(t)$ is heating flowing inward the system; Y_0 is initial temperature; C is the equivalent system thermal capacity; and R is the equivalent thermal resistance between the system borders and surrounding.

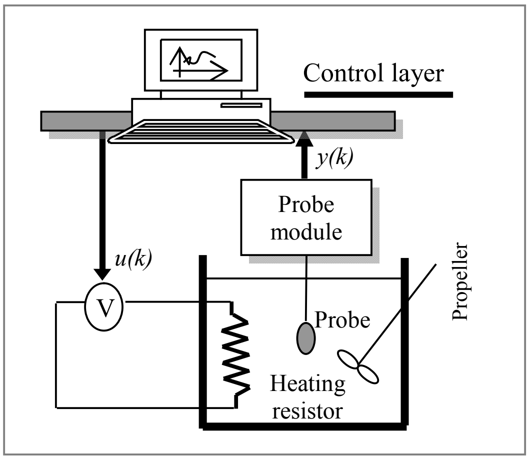

Fig.1. Process Structure

Assuming that R and C are essentially constant, we rewrite the system in Eq. (12) into discrete-time form with some reasonable approximation:

$$y(k+1) = e^{-\alpha T_s}.y(k) + \frac{\frac{\beta}{\alpha}(1-e^{-\alpha T_s})}{1+e^{0.5y(k)-40}}u(k) + (1-e^{-\alpha T_s})Y_0 \qquad (13)$$

Where α and β are some constant values describing R and C. The system parameters used in this example are: $\alpha = 1.0015\times10^{-4}$, $\beta = 8.67973\times10^{-3}$ and $Y_0 = 25.0$ (°C), which were obtained from a real water bath plant in The input $u(k)$ is limited between 0v and 5v. The sampling period is $T_s = 30\,sec$.
In the first part of simulation, the controller objective is to generate a control $u(k)$ allows the system to track the reference trajectory given by:

$$y_{ref} = \begin{cases} 35°C & for \quad k \le 40 \\ 55°C & for \quad 40 < k \le 80 \\ 75°C & for \quad 80 < k \le 120 \end{cases} \qquad (14)$$

The 120 models applies are selected from the input-output characteristic to cover the space Entire benchmark yield.

TABLE I
BA Control parameters values

Parameter	Designation	value
n	Population size	20
ng	Number of generation	100
f_{min}	Minimum frequency	0
f_{max}	Maximum frequency	1
A	Loudness	0.5
r	Pulse rate	0.5

The proposed approach is applied to optimize a fuzzy controller of type Takagi-Sugeno zero-order of water-bath system. As in a previous study [16], the fitness function for performance evaluation is defined to be the sum of absolute error:

$$SAE = \sum_k \left|y_{ref}(k) - y(k)\right| \qquad (15)$$

where $y_{ref}(k)$ and $y(k)$ are the reference output and the actual output of the simulated system, respectively.

The Bat algorithm parameters used in this study are given in Table I.
Figure 2 shows the evolution of the performance index during the execution of the algorithm.

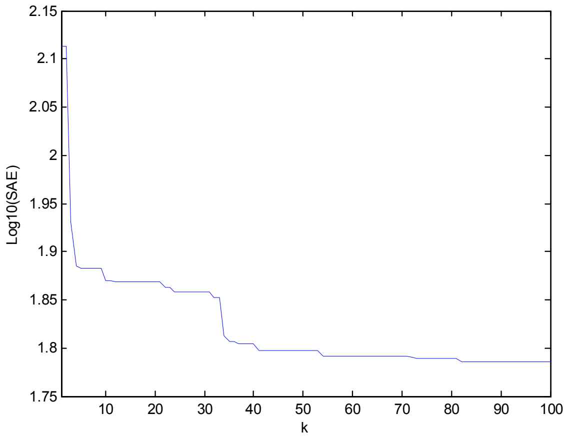

Fig.2. Evolution of the Fitness Function.

The fuzzy rule base obtained after optimization using Bat algorithm is as follows:

$\boldsymbol{R_1}$: $if\ e\ is\ N(-1,-0.55,0.5)\ and\ \Delta e\ is\ N(-1,-0.85,-0.15)\ then\ \Delta u = -1.02$
$\boldsymbol{R_2}$: $if\ e\ is\ Z(-0.55,0.5,0.96)\ and\ \Delta e\ is\ Z(-0.85,-0.15,0.56)\ then\ \Delta u = 0.24$
$\boldsymbol{R_3}$: $if\ e\ is\ P(0.5,0.96,1.00)\ and\ \Delta e\ is\ P(-0.15,0.56,1.00)\ then\ \Delta u = 1.23$

The control system will be calculated by the following formula:

$$u(k) = u(k-1) + G_{\Delta u}\,.\,\Delta u(k) \qquad (15)$$

Where Δu is the command increment, $G_{\Delta u}$ is the scale factor of the variation of control.

The Figure 3 shows the response of the system and the generated command. We can see that the new approach gives a good convergence to the desired trajectory.

To test the robustness of the optimized controller, the trajectory of the reference was changed. We define:

$$y_{ref} = \begin{cases} 34°C & for\ k \le 30 \\ \left(34 + 0.5*(k-30)\right)°C & for\ 30 < k \le 50 \\ \left(44 + 0.8*(k-50)\right)°C & for\ 50 < k \le 70 \\ \left(60 + 0.5*(k-70)\right)°C & for\ 70 < k \le 90 \\ 70\,°C & for\ 90 < k \le 120 \end{cases} \qquad (16)$$

ISSN 2356-5608

The tracking performance of the proposed approach is shown in fig. 4. The results show that the Bat algorithm is able

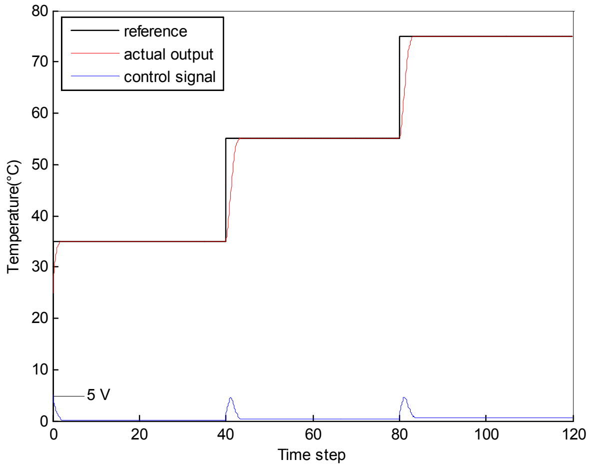

Fig.3. Water bath temperature results for nominal parameters.

to design an optimized fuzzy rule base of fuzzy Takagi-Sugeno controller of type zero order that gives a good tracking of trajectory.

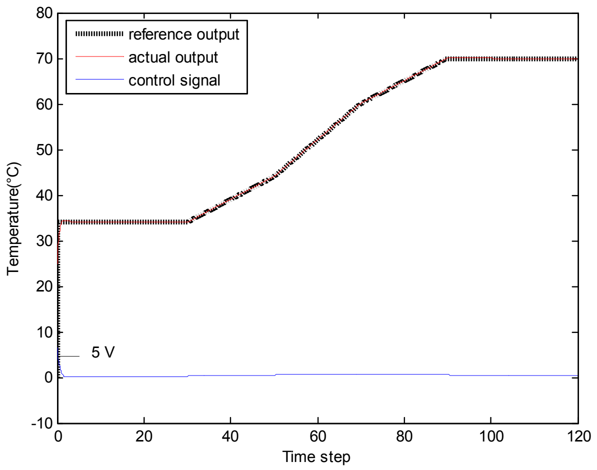

Fig.4. Test of robustness changing the model of reference

TABLE II
COMPARISON OF DIFFERENT METHODS OF OPTIMIZATION FOR WATER BATH TEMPERATURE PROBLEM

Method	Rule number	SAE
ACO[32]	9	130.2
EGA[33]	9	109.7
HGAPSO[34]	9	98.1
RCACO[16]	9	63.6
HEGATS [35]	3	54.42
HPSOTS [36]	3	62.7
BA	3	60.78

Table II shows the results of EGA, ACO, HGAPSO, RCACO, HEGATS and HPSOTS, for the same design problem [16]. These results show that the average error of BA is smaller than the most of the algorithms.

VI. CONCLUSION

This paper proposes a new approach based in using of Bat algorithm for designing fuzzy rule base of fuzzy controller of type Takagi-Sugeno zero order. This algorithm adjusts fuzzy rule parameters automatically, using characteristics of intensification of Bat algorithm. The fuzzy rule base is composed of three rules extracted from analysis, and a triangle membership function for the fuzzification. To demonstrate the effectiveness of the presented approach, a control of water bath temperature is selected. Simulation results show that the proposed approach presented here is a powerful tool for the control of nonlinear systems.

REFERENCES

[1] L. A. Zadeh, “Fuzzy sets”, Information and Control, Vol. 8, 1965, pp. 338-353.

[2] M.S. Ju, D.L. Yang, “Design of adaptive fuzzy control based on natural control laws”, Fuzzy Sets and Systems, Vol. 81, 1996, pp. 191-204.

[3] L.X. Wang and J.M. Mendel, “Back-Propagation Fuzzy System as Nonlinear Dynamic System Identifiers”, *Proceedings of the first IEEE Conference on Fuzzy Systems* , 1992, pp.1409-1418.

[4] S. Horikawa, T. Furuhashi and Y. Uchikawa, “Composition Methods and Learning Algorithms of Fuzzy Neural Networks,” Japanese Journal of Fuzzy Theory and Systems , 4(5), 1992, pp. 529-556.

[5] L.X. Wang, “Stable Adaptive Fuzzy Control of Nonlinear Systems”, *IEEE, Trans. On Fuzzy Systems,* Vol. 1, No. 2, 1993, pp. 146-155.

[6] H. Nomura, I. Hayashi and N. Wakami, “A Learning Method of Fuzzy Inference Rules by Descent Method,” Proceedings of the first IEEE Conference on Fuzzy Systems , 1992, pp. 203-210.

[7] J.S.R. Jang, “ANFIS: Adaptive-Network-Based Fuzzy Inference Systems”, *IEEE Transactions on Systems, Man, and Cybernetics,* 23(3), 1993, pp. 665-685.

[8] Y. Shi and M. Mizumoto, “An improvement of neuro-fuzzy learning algorithm for tuning fuzzy rules”, *Fuzzy sets and Systems*, vol. 118, 2001, pp. 339-350.

[9] I. Rojas, H. Piomares, J. Ortega, A. Prieto, “Self-organized fuzzy system generation from training examples”, *IEEE Trans. Fuzzy Systems*, vol. 1, 2000, pp. 23-36.

[10] M. Mohammadian and R. stonier, “Generating fuzzy rules by genetic algorithms”, *Proceeding of 3rd IEEE international Workshop on Robot and Human Communication,* Nagoya, 1994, pp. 362-367.

[11] C.C. Chen and C.C. Wong, “Self-generating rule-mapping fuzzy controller design using a genetic algorithm”, *In IEE Proceedings on Control Theory and Applications,* vol. 49, 2002, pp. 143–148.

[12] K. Belarbi, F. Titeli, W. Bourebia et K. Benmahammed, “Design of mamdani fuzzy logic controllers with rule base minimisation using genetic algorithm”, *Engineering applications of artificial intelligence*, vol. 18, 2005, pp. 875–880.

[13] N. Talbi and K. Belarbi, “Optimization of Fuzzy Controller using Hybrid Tabu Search and Particle Swarm Optimization”, *in Proc. IEEE, 11th International Conference on Hybrid Intelligent Systems (HIS)*, Malaysia, 2011, pp. 561-565.

[14] V. Mukherjee., .P. Ghoshal, “Intelligent particle swarm optimized fuzzy PID controller for AVR system”. Electric Power Systems Research, Vol. 77, 2007, pp. 1689-1698.

[15] N. Talbi and K. Belarbi, “Designing Fuzzy Controllers for a Class of MIMO Systems using Hybrid Particle Swarm Optimization and Tabu

ISSN 2356-5608

Search", *International Journal of Hybrid Intelligent Systems, IOS Press,* vol. 10, n°1, 2013, pp. 1–9.

[16] C. F. Juang, P. H. Chang, "Designing fuzzy-rule-based systems using continuous Ant colony optimization", IEEE trans., *Fuzzy Syst.*, vol. 18, no. 1, 2010, pp. 138-149.

[17] X.S.Yang and A. H. Gandomi, "Bat Algorithm: A Novel Approach for Global Engineering Optimization", Engineering Computations, Vol. 29, Issue 5, 2012, pp. 464--483.

[18] J. Kennedy et R.C. Eberhart, " Particle swarm optimization", *IEEE International Conference on Neural Networks,* Piscataway, 1995, pp. 1942-1948.

[19] S. Kirkpatrick, C.D. Gelatt, and M.P. Vecchi, "Optimization by simulated annealing", Science, vol. 220,1983, pp. 671-680.

[20] Z.W. Geem, J. Kim, G.V Loganathan, "A new heuristic optimization algorithm: Harmony search", Simulation, vol. 76,2001, pp. 60-68.

[21] X.S. Yang, "Nature-inspired Metaheuristic Algorithms", Luniver Press, 2008.

[22] X.S. Yang, "A New Metaheuristic Bat-Inspired Algorithm", Nature Inspired Cooperative Strategies for Optimization (NISCO 2010) (Eds. J. R. Gonzalez et al.), Studies in Computational Intelligence, Springer Berlin,Springer,vol.284, 2010, pp. 65-74.

[23] K. Khan, A. Nikov,A. Sahai, "A fuzzy bat clustering method for ergonomic screening of office workplaces", *Advances in Intelligent and Soft Computin,* Springer , Vol. 101, 2011.; pp. 59–66.

[24] G.Q. Huang,W-J Zhao, Q.Q Lu, "Bat algorithm with global convergence for solving large-scale optimization problem", *Application Research of Computers*, 30(5), 2013, pp. 1323–1328.

[25] J. Zhang, G. Wang. "Image matching using a bat algorithm with mutation", *Applied Mechanics and Materials*, 203(1), 2012, pp.88–93.

[26] A. Alihodzic, M. Tuba, "Improved Hybridized Bat Algorithm for Global Numerical Optimization", *in Proc. IEEE, 16th International Conference on Computer Modelling and Simulation, 2014, pp.57-62.*

[27] C. C. Lee, "Fuzzy logic in control systems: fuzzy logic controller. Part 1", *IEEE Transactions on systems Man and Cybernetics*, Vol. 20, 1990, pp. 404-418.

[28] A.H. Gandomi, X.S. Yang, A.H. Alavi, and S. Talatahari, Bat algorithm for constrained optimization tasks, *Neural Computing & Applications*, pages 1–17, 2012.

[29] P.W. Tsai, J.S. Pan, B.Y. Liao, M.J. Tsai, and V. Istanda. Bat algorithm inspired algorithm for solving numerical optimization problems, *Applied Mechanics and Materials*, 148:134–137,2012.

[30] X.S. Yang. Review of meta-heuristics and generalised evolutionary walk algorithm, *International Journal of Bio-Inspired Computation*, 3(2):77–84, 2011.

[31] J. Tanomaru, S. Omatu, process control by on-line trained Neural controllers, *IEEE trans. , Industrial Electronics*, vol. 39, N° 6, 1992, pp. 511-521.

[32] C.F. Juang, "A TSK-type recurrent fuzzy network for dynamic systems processing by neural network and genetic algorithms," *IEEE Trans. Fuzzy Syst.*, vol. 10, no. 2, Apr. 2002, pp. 155–170.

[33] C. F. Juang, C. M. Lu, C. Lo, and C. Y. Wang, "Ant colony optimization algorithm for fuzzy controller design and its FPGA implementation,"*IEEE Trans. Ind. Electron.*, vol. 55, no. 3, Mar. 2008, pp. 1453–1462.

[34] C. F. Juang, "A hybrid of genetic algorithm and particle swarm optimization for recurrent network design," *IEEE Trans. Syst., Man, Cybern. B,Cybern.*, vol. 34, no. 2, Apr. 2004, pp. 997–1006.

[35] N. Talbi and K. Belarbi, "Designing fuzzy rule base using HEGATS: Application for control and modeling", *International Journal of Hybrid Intelligent Systems,* vol. 10, 2013, pp.205-214.

[36] N. Talbi and K. Belarbi, "Fuzzy takagi Sugeno System Optimization using HPSOTS algorithm", *International Journal of of Tomography and Simulation*, vol.22, N°1, 2013, pp. 4-16.

ISSN 2356-5608