World Academy of Science, Engineering and Technology
International Journal of Mechanical and Mechatronics Engineering
Vol:8, No:5, 2014

Half-Circle Fuzzy Number Threshold Determination via Swarm Intelligence Method

P.-W. Tsai, J.-W. Chen, C.-W. Chen, C.-Y. Chen

***Abstract*—**In recent years, many researchers are involved in the field of fuzzy theory. However, there are still a lot of issues to be resolved. Especially on topics related to controller design such as the field of robot, artificial intelligence, and nonlinear systems etc. Besides fuzzy theory, algorithms in swarm intelligence are also a popular field for the researchers. In this paper, a concept of utilizing one of the swarm intelligence method, which is called Bacterial-GA Foraging, to find the stabilized common *P* matrix for the fuzzy controller system is proposed. An example is given in in the paper, as well.

***Keywords*—**Half-circle fuzzy numbers, predictions, swarm intelligence, Lyapunov method.

I. Introduction

ROBOTS with intelligent control can be attributed to the development of fuzzy theory and other artificial intelligence methods. Fuzzy logic linguistic approaches have been widely studied and applied in solving practical problems in engineering and in management since Zadeh first proposed fuzzy theory in 1965 [1]. After several years' development of fuzzy theory, Laarhoven and Perdrycz proposed the triangular fuzzy numbers in 1983 [2]. Although triangular fuzzy numbers are easy and convenient to use, they still require complicated calculations in prediction [3]-[5]. To simplify the complicated operations, the half-circle fuzzy number (HCFN) is proposed [6], [7]. The HCFN is simpler and more convenient when it is employed in trigonometric functions. It can also be used with polar coordinates in integral or other difficult operations used for prediction.

Besides fuzzy theory, swarm intelligence method is also a popular research field in artificial intelligence. Different from fuzzy theory, algorithms in swarm intelligence utilize the tinny intelligence, which takes place in the creatures' behavior or their special strategy for surviving in Mother Nature, to construct a serious of process for solving optimization problems in engineering, management, or economics. For example, by taking felid as the model, Chu et al. propose Cat Swarm Optimization (CSO) [8], [9] for solving numerical optimization problems; Pan et al. pro-pose Fish Migration Optimization (FMO) [10] in 2010, and Tsai propose Evolved Bat Algorithm (EBA) [11] in 2012. Moreover, swarm intelligence methods can be employed to solve engineering problems in two ways: the first is to describe the problem in mathematic formula and exploit the swarm intelligence algorithm, directly, to find the optimum solution. [3], [12]-[16] Another way is to combine swarm intelligence algorithms with other existing methods or systems such as Artificial Neural Network (ANN) [5], [17], [18] and Simulated Annealing [19]. In this paper, we present a concept of employing swarm intelligence algorithm to construct HCFN for the further usage in fuzzy controller systems.

The rest of this paper is constructed as follows: The brief review on triangular fuzzy numbers, HCFN, and the concept of Bacterial-GA Foraging, which is one of the hybrid swarm intelligence algorithm, are given in Section II. Our proposed method is given with an example in Section III. Finally, the conclusion is given in Section IV.

P.-W. Tsai, J.-W. Chen, C.-W. Chen, and C.-Y. Chen are with Department of Maritime Information and Technology, National Kaohsiung Marine University, Kaohsiung, Taiwan (e-mail: chengwu@webmail.nkmu.edu.tw).

C.-W. Chen is also with Faculty of Engineering, King Abdulaziz University, Jeddah 21589, Saudi Arabia.

II. Literature Review

In Mendel's 1995 report, membership functions were for the most part, associated with terms that appeared in the antecedents or consequents of rules, or in phrases. The membership functions used in the fuzzy theory can be constructed in different forms. In other words, the membership functions can be quite different for different users because the users can design the membership functions base on their own experience. It means that two fuzzy systems designed for the same purpose may contain completely different membership functions. However, the most commonly used shapes for membership functions are triangular, trapezoidal, piecewise linear or Gaussian. The fuzzy inference can be depicted in three major parts: quantification, fuzzification, and defuzzification. The membership functions are employed in the fuzzy inference.

A. Triangular Fuzzy Numbers

Laarhoven and Pedrycz proposed the triangular fuzzy numbers in 1983 [20], [21]. It can be defined as follows:

Definition 1: A fuzzy number M on $\boldsymbol{R} = (-\infty, +\infty)$ is a triangular fuzzy number if its membership function $\mu_M: \boldsymbol{R} \to [0, 1]$ is equal to (1):

$$\mu_M(x) = \begin{cases} \frac{1}{m-l}x - \frac{l}{m-l}, & x \in [l, m] \\ \frac{1}{m-u}x - \frac{u}{m-u}, & x \in [m, u] \\ 0, & \text{otherwise} \end{cases} \quad (1)$$

where $l \leq m \leq u$, l and u represent the lower and upper values in support of M, respectively, and m stands for the model value. The triangular fuzzy number, as given by (1), can be denoted by (l, m, u). The hypothesis of M is given as a set of elements $\{x \in \boldsymbol{R} | l < x < u\}$. An example of the membership function of a triangular fuzzy number is given in Fig. 1:

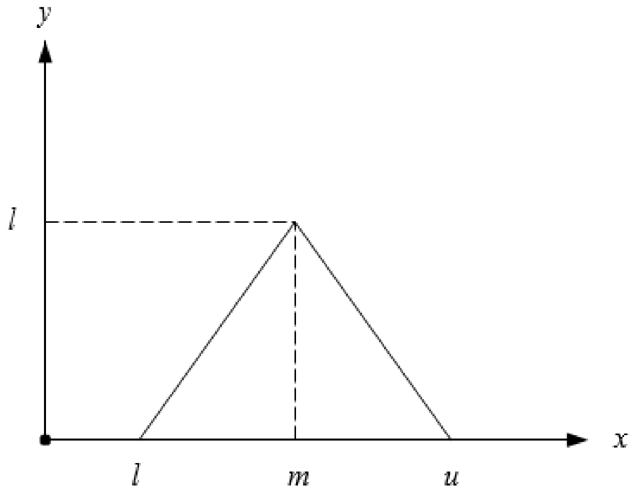

Fig. 1 Membership function of a triangular fuzzy number

B. Half-Circle Fuzzy Numbers

A model of half-circle fuzzy numbers developed based on triangular fuzzy numbers is described as follows:

Definition 2: A fuzzy number H is defined on $\boldsymbol{R} = (-\infty, +\infty)$ as a half-circle fuzzy number if its membership function $\mu_H: \boldsymbol{R} \rightarrow [0, 1]$ is equal to (2):

$$\mu_H(x) = \begin{cases} \sqrt{1-(x-h)^2}, & x \in [h-1, h+1] \\ 0 & , \text{otherwise} \end{cases} \quad (2)$$

Fig. 2 shows an example of the membership function of a half-circle fuzzy number:

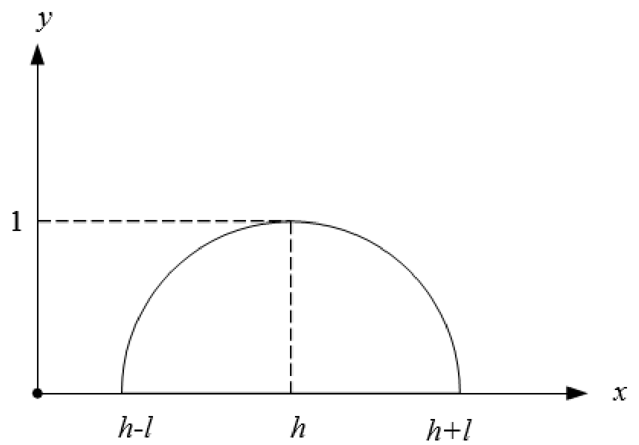

Fig. 2 Membership function of a half-circle fuzzy number

C. Bacterial-GA Foraging Optimization

Swarm intelligence is an artificial intelligence technique based on the study of collective behavior [22]. Many of the algorithms in swarm intelligence are developed based on simulating the behaviors of the creatures in the Mother Nature. Swarm intelligence is symbolically made up of a population of simple agents interacting locally with one another and with their environment. Even though there is no centralized control structure indicating how individual agents should behave, local interactions between such agents often lead to the emergence of global behavior.

Both Bacterial Foraging (BF) [23]-[25] and Genetic Algorithm (GA) [24], [26] are well-known popular algorithms for solving optimization problems. BF simulated the life cycles and the foraging behaviors of bacterial, which are called E. coli, to solve the optimization problems, and GA uses chromosomes to represent the solution sets in the algorithm. However, these algorithms still have their own weaknesses on finding the solutions due to the characteristics of their own evolutional models. In 2000, Kevin M. Passino proposed the idea of BF [24] for solving optimization problems. The framework of BF was based on parroting the behaviors of bacterium, i.e., the way they search nutrients, evade noxious environments, and the moving circumstance. By way of imitating the existence of the bacterium, the optimization problem can be solved.

Chen et al. proposed Bacterial-GA Foraging [27] in 2007. According to their observation, they noticed that BF has higher local search ability than GA, but it is poor in global search while the solution space is huge. On the contrary, GA performs high capacity on global search, but it is difficult for GA to pinpoint the global best solution. Hence, they designed a hybrid structure to combine these two algorithms and utilized it to solve problems in optimization. The flowchart of Bacterial-GA Foraging is shown in Fig. 3.

Assume that we are going to minimize the output of a *D*-dimensional fitness function. The solution in the solution space can be denoted by (3).

$$\theta = [x_1, x_2, x_3, \dots, x_D] \quad (3)$$

where θ is the coordinate of a solution in the solution space in the *D*-dimensional solution space, and x stands for the value on the corresponding dimension. Thus, the operation of Bacterial-GA Foraging can be depicted as follows:

- **Initialization**: Randomly spread the bacteria into the solution space by randomly assign values for the coordinates in (3).
- **Evaluation**: Evaluate the fitness values of the bacteria by the fitness function designed by the user. Since we're now minimizing the output of the fitness function, the artificial agent with the smallest fitness value stands for the near best solution in this iteration.
- **Movement (Tumbling/Swimming)**: Every bacteria in the population will tumble once, and the bacteria, which have better fitness values after the tumble process, will further go into the swimming process under the swim length limitation N_s. The tumbling process is operated by (4)

$$\theta^i(j+1, k, l) = \theta^i(j, k, l) + C(i)\phi(j) \quad (4)$$

where i, j, k and l are the indices of the bacterium, the chemo-tactic step, the reproduction step, and the elimination-dispersal event, respectively. $C(i)$ is the maximum step size for the bacterium to move, and $\phi(j)$ denotes a random variable vector in the range of [0, 1]. Swimming process takes exactly the same operation as tumbling. It takes place in the iteration when the trigger condition is satisfied. In addition, the maximum repeat of swimming process is limited by N_s.

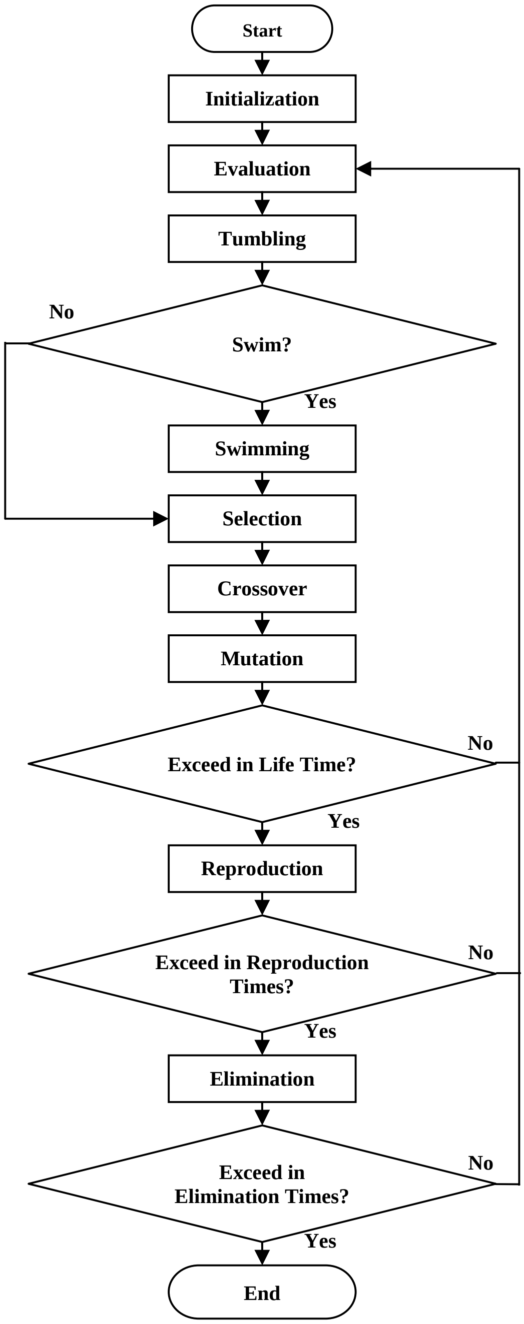

Fig. 3 The flowchart of Bacterial-GA Foraging

- **Selection**: According to the selection rate, part of the chromosomes will be eliminated, and the other part of them will be kept. There are lots of methods to select some chromosomes to keep. And the most popular ways are random selection, roulette wheel selection, and essential selection.
- **Crossover**: This step forces the chromosomes to exchange the information they carried to the others via switching the information on parts of the dimensions. The popular crossover methods are one-site crossover, and two-site crossover.
- **Mutation**: According to the mutation rate, several chromosomes will be selected from the population, and then randomly change the value of parts of the dimensions. This will give the population a larger chance to generate new species. For optimization, it is a chance to get an abrupt evolution.
- **Reproduction**: In the process of reproduction, the whole population will only keep half of the bacteria, which perform better healthy values, and directly reproduce them to replace the bacteria with worse healthy values.
- **Elimination**: When the reproduction times exceed the limit which user defined, the elimination-dispersal even takes part in the process. This process eliminates the bacteria with probability, if a bacterium is eliminated, a bacterium, which fits in with the initial conditions, is generated to replace the eliminated one.

III. Our Proposed Method

To employ swarm intelligence methods solving problems of optimization, a fitness function is required at the first beginning. The fitness function is the mathematic representation of the evaluation condition for the target problem. In our design, we're going to use Bacterial-GA Foraging to find a common P matrix, which satisfies the condition listed in (5):

$$(A_i - B_i K_l)^T P + P(A_i - B_i K_l) < 0 \tag{5}$$

where $P = P^T > 0$ and $i, l = 1,2, \ldots, D$.

According to Hsiao et al.'s report [14], the equilibrium point of a closed-loop fuzzy system is asymptotically stable in the large, if there exists a common positive definite matrix P, which satisfies (5).

The objective of our proposed method is to choose the proper common matrix P for the half-circle membership functions which satisfy (5). An example is given as follows: Consider a nonlinear system described by (6) [13]:

$$\begin{cases} x_{11}(t) = -29x_{11}(t) + x_{21}(t) - 0.5x_{21}^2(t) \\ x_{21}(t) = 3x_{11}(t) - 12x_{21}(t) - 0.5x_{21}^2(t) + u(t) \end{cases} \tag{6}$$

Step 1. We establish a T-S fuzzy model for the above system and the nonlinear system (6) can be approximated by the following fuzzy models:

Rule 1: IF $x_1(t)$ is M_{11}, THEN $x(t) = A_1 x_1(t) + B_1 u_1(t)$;
Rule 2: IF $x_1(t)$ is M_{21}, THEN $x(t) = A_2 x_1(t) + B_2 u_1(t)$;

where $x_1(t) = [x_{11}(t)\, x_{21}(t)]^T$, $A_1 = \begin{bmatrix} -29 & 1 \\ 3 & -12 \end{bmatrix}$, $A_2 = \begin{bmatrix} -29 & 0.5 \\ 3 & -12.5 \end{bmatrix}$, $B_1 = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$, $B_2 = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$. The half-circle membership functions for Rule 1 and 2 are plotted in Fig. 4.

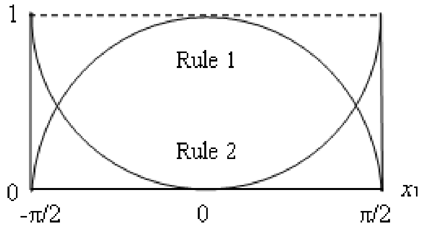

Fig. 4 The membership functions of HCFN

Step 2. Utilizing Bacterial-GA Foraging to find the common P matrix, which satisfies the stabilization condition listed in (5). The fitness function in our design is a Boolean combination of two conditions. Since the stabilization condition has two requirement, the fitness function is designed in (7):

$$F = \alpha \times \beta \tag{7}$$

where F denotes the fitness value, and $\times$ stands for the AND operation in Boolean logic; α and β come from (8) and (9).

$$\alpha = \begin{cases} 1, & \text{if } (A_i - B_i K_l)^T P + P(A_i - B_i K_l) < 0 \\ 0, & \text{otherwise} \end{cases} \tag{8}$$

$$\beta = \begin{cases} 1, & P = P^T > 0 \\ 0, & \text{otherwise} \end{cases} \tag{9}$$

IV. Discussion and Conclusion

In this paper, we use Baterial-GA Foraging algorithm to find the feasible solutions of the common P matrix, which satisfies the stabilization conditions for the fuzzy controller. One of the conventional ways to find the common P matrix is utilizing the Linear Matrix Inequalities (LMI) tool. As the same as the swarm intelligence methods, the LMI requires the recursive computing. The LMI reports one solution for the corresponding input system parameters at last. However, the swarm intelligence methods, like the Bacterial-GA Foraging, we use in this study has the capacity to report multi feasible solutions at once.

On the other hand, both the HCFN and the triangular fuzzy numbers are easy to use in the fuzzy controllers. In this paper, we choose HCFN to be the membership function. In the future, we plan to employ the swarm intelligence algorithms in the fuzzy controllers with the triangular membership functions, as well. The result is predicable to be positive.

References

[1] A. Bogomolny, "On the perimeter and area of fuzzy sets, Fuzzy Sets and Systems," vol. 23, 1987, pp. 257-269.
[2] P.J.M. van Laarhoven and W. Perdrycz, "A fuzzy Extension of Saaty priority theory," Fuzzy Set and Systems, vol. 11, 1983, pp. 229-241.
[3] D. Dubois and H. Prade, "Fuzzy Sets and Systems: Theory and Applications," Academic Press, New York 1980.
[4] H. J. Zimmermann, Fuzzy Sets, Decision Making and Expert Systems, (1991) Boston: Kluwer Academic Publisher.
[5] H. J. Zimmermann., Fuzzy Set Theory and Its Applications, (1991) Kluwer, Dordrecht.
[6] C. Y. Chen, W. I. Lee, Y. C. Sui, and C. W. Chen, "Applying half-circle fuzzy numbers to control system: a preliminary study on development of intelligent system on marine environment and engineering," World Academy of Science, Engineering and Technology, vol. 72, 2010, pp. 204-209.
[7] F.H. Hsiao, W.L. Chiang, C.W. Chen, S.D. Xu and S.L. Wu, "Application and robustness design of fuzzy controller for resonant and chaotic systems with external disturbance," International Journal of Uncertainty, Fuzziness and Knowledge-Based System, vol. 13, 2005, pp. 281-295.
[8] S.-C. Chu, and P.-W. Tsai, "Computational Intelligence Based on the Behavior of Cats," International Journal of Innovative Computing, Information and Control, vol. 3, no. 1, 2007, pp. 163-173.
[9] S.-C. Chu, P.-W. Tsai, and J.-S. Pan, "Cat Swarm Optimization," in *Proc.* 9th Pacific Rim International Conference on Artificial Intelligence, LNAI 4099, 2006, pp. 854-858.
[10] J.-S. Pan, P.-W. Tsai, and B.-Y. Liao, "Fish Migration Optimization Based on the Fishy Biology," in *Proc.* of 2010 4th International Conference on Genetic and Evolutionary Computing, 2010, pp. 783-786.
[11] 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, vol. 148-149, 2012, pp. 134-137.
[12] C.W. Chen, W.L. Chiang, and C.H. Tsai et al., "Fuzzy Lyapunov method for stability conditions of nonlinear systems," International Journal on Artificial Intelligence Tools, vol. 15, 2006, pp. 163-171.
[13] C.W. Chen, W.L. Chiang, and F.H. Hsiao, "Stability analysis of T-S fuzzy models for nonlinear multiple time-delay interconnected systems," Mathematics and Computers in Simulation, vol. 66, 2004, pp. 523-537.
[14] F.H. Hsiao, C.W. Chen, and Y.W. Liang et al., "T-S fuzzy controllers for nonlinear interconnected systems with multiple time delays," IEEE Trans. Circuits & Systems-I, vol. 52, 2005, pp. 1883- 1893.
[15] W. I. Lee, C. W. Chen, C. Y. Chen, and Y. C. Sui, "Application of half-circle fuzzy numbers and development of triangular fuzzy numbers to fuzzy control," in *Proc.* 10th WSEAS International Conf. on Applied Computer Sciences (ACS'10), Japan, 2010, pp. 335-340.
[16] J.M. Mendel, "Fuzzy logic systems for engineering: A tutorial," in *Proc.* of the IEEE, 83 March 1995.
[17] J. J. Buckley, "Fuzzy Hierarchical Analysis," Fuzzy Sets and System, vol. 17, 1985, pp. 233-247.
[18] C.W. Chen, C.L. Lin, C.H. Tsai, C.Y. Chen, and K. Yeh, "A novel delay-dependent criteria for time-delay T-S fuzzy systems using fuzzy Lyapunov method," International Journal on Artificial Intelligence Tools, vol. 16, 2007, pp. 545-552.
[19] J. J. Buckley, "Ranking alternatives using fuzzy number," Fuzzy Set and System, vol. 15, 1985, pp. 21 -31.
[20] J. J. Buckley and E. Eslami, "Fuzzy plane geometry I: Points and lines," Fuzzy Sets and Systems, vol. 86, 1997, pp. 179-187.
[21] J.J. Buckley and E. Eslami, "Fuzzy plane geometry II: Circles and polygons," Fuzzy Sets and Systems, vol. 87, 1997, pp. 79-85.
[22] G. Beni, J. Wang, "Swarm Intelligence in Cellular Robotic Systems," in *Proc.* of NATO Advanced Workshop on Robots and Biological Systems, Tuscany, Italy, 1989.
[23] S. Mishra, and C. N. Bhende, "Bacterial Foraging Technique-Based Optimized Active Power Filter for Load Compensation", IEEE Transactions on Power Delivery, vol. 22, no. 1, 2007, pp. 457-465.
[24] Kevin M. Passino, "Biomimicry of Bacterial Foraging for Distributed Optimization and Control," Control Systems Magazine, June 2002, pp. 52-67.
[25] W. J. Tang, Q. H. Wu, and J. R. Saunders, "Bacterial Foraging Algorithm For Dynamic Environments," in *Proc.* IEEE Congress on Evolutionary Computation, IEEE, July 2006, pp. 1324-1330.
[26] J. S. Pan, F. R. McInnes and M. A. Jack, "Application of Parallel Genetic Algorithm and Property of Multiple Global Optima to VQ Codevector Index Assignment," Electronics Letters, Vol. 32, No. 4, 1996, pp.296-297.
[27] T.-C. Chen, P.-W. Tsai, S.-C. Chu, and J.-S. Pan, "A Novel Optimization Approach: Bacterial-GA Foraging," in *Proc.* 2nd International Conf. on Innovative Computing, Information and Control (ICICIC-2007), Japan.