SSRG International Journal of Electrical and Electronics Engineering Volume 11 Issue 8, 217-225, August 2024
ISSN: 2348-8379/ https://doi.org/10.14445/23488379/IJEEE-V11I8P119

Original Article

A New Approach to Automatic and Optimal Membership Function Generation for Fuzzy System Modelling

Anup Kumar Mallick[1], Dwaipayan Ghosh[2], Kabita Purkait[3]

[1,2,3]*Department of Electronics & Communication Engineering, Kalyani Government Engineering College, West Bengal, India.*

[1]*Corresponding Author : anup.mallick@kgec.edu.in*

Received: 09 June 2024 Revised: 11 July 2024 Accepted: 10 August 2024 Published: 31 August 2024

Abstract *- During the last few decades, the optimization-based data-driven approach has been widely used for generating membership functions in fuzzy-based systems, where the shapes of membership functions are mostly considered either triangular or trapezoidal. However, the number of parameters that are required to be estimated for a triangular membership function is three (left vertex, center, and right vertex). For a trapezoidal membership function, it is four (left base point, left shoulder, right base point, and right shoulder). Whereas, the number of parameters required for a Gaussian membership function is two (mean and standard deviation). Therefore, a fuzzy system modelled using the Gaussian membership function can significantly reduce the number of parameters when the number of subsets for the antecedent and consequent membership functions is large. However, not much attention is given to designing fuzzy models with Gaussian-shaped membership functions; most of the existing fuzzy modelling techniques impose many restrictions on the membership functions' parameters. As a result, the flexibility and scope of the optimization techniques are reduced. This study, therefore, suggests a novel optimization-based technique to frame fuzzy membership functions in which the membership functions are Gaussian-shaped, and very few restrictions are imposed on the parameter selection. A comparative analysis is carried out between the conventional method and the proposed method with different optimization techniques (Differential Evolution (DE), Particle Swarm Optimization (PSO), and Genetic Algorithm (GA)) to approximate four standard nonlinear functions.*

Keywords *- Differential Evolution, Fuzzy system, Genetic Algorithm, Membership Function, Particle Swarm Optimization.*

1. Introduction

The basic principles of fuzzy modelling were formulated by Zadeh [1] with the aim of approximately but effectively describing the behaviours of complex or ill-defined systems. In the last few decades, fuzzy models have been widely used in different fields of science, engineering, social science, medical diagnosis and treatment, etc. [2-4]. There are two broad types of fuzzy models, viz., the Mamdani fuzzy model and the Takagi-Sugeno fuzzy model. The Mamdani model is a linguistic model that is based on a collection of IF-THEN rules, with the antecedent and the consequent both fuzzy [5]. The Sugeno models are formed by rules with fuzzy antecedent and functional consequent [6]. In this paper, the experiment is conducted on the modelling of the Mamdani-type fuzzy system.

The major tasks involved in modelling a Mamdani-type fuzzy-based system are the generation of fuzzy Membership Functions (MFs) for the antecedent and consequent, formation of the rule base, development of a fuzzy inference engine, and finally, the defuzzification for the crisp output [7, 8]. The construction of membership functions plays a pivotal role in the design process of a Mamdani-type fuzzy model. One conventional way to formulate membership functions is to divide the input and output spaces equally to generate the antecedent and consequent membership functions, respectively.

Another method of modelling fuzzy systems is to use expert knowledge. However, knowledge-based membership function generation has some limitations. Sometimes, the expert in the particular domain is not available, or the experts' opinions may differ from one another. Therefore, researchers [9-11] have developed different approaches, such as clustering-based approach [12-14], neural network-based approach [15], density-based technique [16], statistical approach [17, 18], and optimization-based method [19-23], etc., to derive the fuzzy models' membership functions.

In the last few decades, optimization techniques, due to their ability to automatically generate the optimum value of parameters, have been extensively used for the generation of MFs of fuzzy systems. In such an approach, the optimization techniques are used either to fine-tune the membership functions after initial guesses or to generate the membership functions automatically. Different optimization techniques, viz., GA, PSO, and DE, have been used to generate the membership functions optimally.

Karr et al. [19] have employed GA to model a fuzzy controller for balancing a cart pole. Herrera et al. [20] have used a genetic algorithm to tune fuzzy rules and fit fuzzy membership functions for balancing an inverted pendulum. The membership functions for a single input-single output system where the output points are squares of the input points are determined using a genetic algorithm in [21]. Zhang et al. [22] optimize the membership function for a general industrial process with dead time, saturation, and time delay using a genetic algorithm. Safaee et al. [23] apply PSO and GA to generate the membership functions in the design of a quad rotor.

Although several research findings claim the efficacy of the optimized fuzzy systems over the conventional fuzzy systems, the major problem in the optimum generation of membership functions is the restrictions imposed on the parameters, which in turn reduces the search space for the optimization problem. For example, in [20], the authors update a fuzzy number by four parameters, and the formulation of these four parameters sufficiently restricts the parameters of membership functions.

Zhang et al. [22] put several constraints on the position of membership functions. They [22] consider triangular-shaped membership functions and restrict the position of the centers and the two vertexes of the membership functions. As the number of constraints increases, the flexibility and scope of generating new positions for the parameters for the membership functions reduces. Moreover, it is found that most of the optimization-based MF generation techniques have considered the shape of the MFs, either triangular [21-23] or trapezoidal [20], whereas both shapes require more parameters to be selected for their formation.

Motivated by the above-discussed problems in membership function generation, this paper suggests a novel technique for constructing MFs for the antecedent and consequent in a Mamdani-type fuzzy system. The shapes of the membership functions of both the antecedent and consequent are considered as Gaussian. The performances of the proposed model with varying optimization techniques (DE, PSO, and GA) are compared with the conventional fuzzy system modelling method in predicting four standard nonlinear functions, viz., a cube function, a square function, a square root function, and an exponential function.

2. Theoretical Preliminaries

The presented paper devises an optimization-based membership function generation technique for a Mamdani-type fuzzy system. Hence, basic concepts of a Mamdani-type fuzzy system, fuzzy membership functions, and optimization techniques are discussed briefly in this section.

2.1. Mamdani-Type Fuzzy System

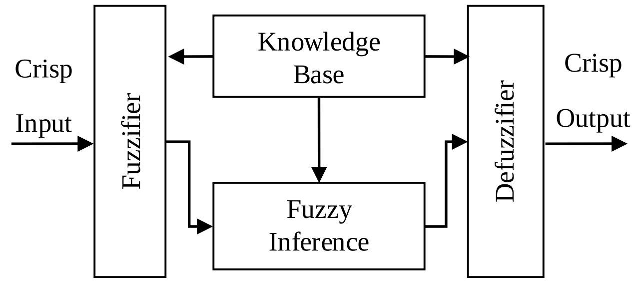

Fig. 1 Block diagram of a Mamdani-type fuzzy system

Figure 1 shows the block diagram of a Mamdani-type fuzzy system. It consists of a fuzzifier, a knowledge base, a fuzzy inference engine, and a defuzzifier. The knowledge base comprises a rule base and a database. The rule base consists of a set of fuzzy If-Then rules, and the database keeps the MFs of the input and output variables. The fuzzifier, with the help of antecedent MF, converts real-world crisp input to its corresponding fuzzy value. The fuzzy inference engine fires appropriate rules of the rule base with varying firing strength and provides fuzzy outputs of each fired rule. The defuzzifier generates a crisp output from the aggregated FIS output for real-world applications using the defuzzification method.

2.2. Mamdani-type Fuzzy System

A membership function $\mu_A(x)$ is defined by the following mapping:

$$\mu_A : x \rightarrow [0,1], \quad x \in X. \tag{1}$$

Where x is a real number describing an object, X is the universe of discourse, and A is a subset of X. The membership function in fuzzy logic maps an attribute or object to a positive real number in the interval [0, 1]. Because of its function-like mapping characteristics, it is called a membership function [24].

The membership functions used for mapping input variables and output variables are termed antecedent membership function and consequence membership function, respectively. Theoretically, any function can serve as a membership function for a given fuzzy set [25]. The shape of the membership function depends on the context of the applications. Different methods of generating membership functions have been proposed in the literature, some of which are outlined in the introduction section of this article.

2.3. Optimization Techniques

Optimization is a technique that aims to maximize or minimize a function in the design of a system. The function, here, termed a cost or objective function, attempts to fulfil some performance specifications or targets. The optimization techniques return a set of parameters' values to obtain the best possible result for the design of the underlying system.

There are several optimization techniques or algorithms, such as GA, PSO, DE, etc. GA relies on the possibility of generating better children from fitter parents and the survival of the fittest. PSO is inspired by the social movement of organisms in a bird flock or fish school.

Differential evolution [28–30] is a search algorithm that optimizes a problem by iteratively developing a candidate solution through an evolutionary process with little or no assumption about the underlying optimization problem and is capable of rapidly exploring large design areas. The optimization techniques are used for various purposes; optimal generation of membership functions in a fuzzy system is one such application.

3. Proposed Model

The design of the proposed model broadly involves four major steps: generation of fuzzy MFs, formation of a fuzzy rule base, rule firing by the inference engine, and defuzzification of the fuzzy output. The novelty of the suggested method lies in its first step, i.e., the generation of the fuzzy MFs, which is thus illustrated in detail. The other three steps, viz., formation of fuzzy rules, the firing of rules by the fuzzy inference engine, and finally, defuzzification, are done in the conventional methods of fuzzy system design; hence, these three steps are described in brief.

3.1. Generation of Fuzzy Membership Functions

The proposed model is designed for a single-input, single-output system. The input variable represented by antecedent MF and the output variable represented by consequent MF are both comprised of three fuzzy subsets: low, medium, and high, and the MFs are assumed to be Gaussian-shaped. The minimum and maximum values of the variables are considered as the mean of low subset and high subset, respectively, for both antecedent and consequent. The mean of the subset middle and the Standard Deviations (SDs) of all three subsets for the antecedent and the consequent are determined using an optimization technique. In the evolutionary algorithm, the composition of an individual is given in Figure 2.

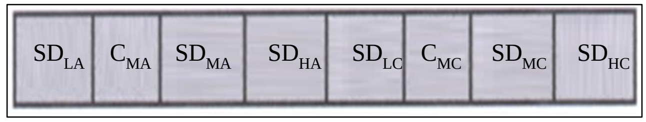

Fig. 2 Individual representing the membership functions' parameters

Here, SD_{LA}, SD_{MA}, SD_{HA} denote SDs of the MFs representing the input variable for its low, medium, and high subsets, respectively; C_{MA} and C_{MC} represent the medium subset's mean values for the input variable and the output variable, respectively; and SD_{LC}, SD_{MC}, and SD_{HC} are the notions of SDs for the low, medium, and high subsets of the output variable, respectively. The optimization algorithm in each iteration aims to reduce the objective function given in Equation (1).

$$f = \sum_{i=1}^{N} ((g(i) - d(i))^2 \quad (2)$$

Here, N represents the number of data points, *g(i)* and *d(i)* denote, respectively, the given output and the derived output obtained using the proposed technique for the *i*-th input *x(i)*. The best population found at the last iteration of the optimization technique is translated as membership functions' parameters.

This paper proposes and compares the generation of membership functions' parameters using three different evolutionary algorithms (GA, PSO, and DE). The procedures for generating the parameters of MFs using GA, PSO, and DE are given in Algorithms 1, 2, and 3, respectively.

Algorithm 1. Parameter optimization of member functions using GA
Input: Population size (*n*), maximum number of iterations (*itrmax*), crossover, and mutation probabilities.
Output: Membership functions' parameters.
1: Generate initial population or parents (*nx8*)
2: Set iteration=0
3: Compute the fitness of the parents using objective function in Equation (1)
4: *while* iteration < *itrmax do*
5: Increment iteration by 1
6: Form the mating pool from the parents through binary tournament selection
7: Perform crossover & mutation
8: Compute the fitness of the offspring using objective function in Equation (1)
9: Update parents for the next generation by taking the best fit *n* chromosomes among the parents & the children
10: end while
Return best chromosome

Algorithm 2. Parameter optimization of member functions using PSO

Input: Swarm size (*n*), maximum number of iterations (*itrmax*), *w, c1, c2*.

Output: Membership functions' parameters.

1: Generate initial particles (*nx8*) and initial velocity(old)

2: Set iteration=0

3: Compute the fitness of each particle using objective function in Equation (1)

4: Assign the current position as the *pbest* of each particle

5: Assign the best position of all particles as the *gbest*

6: *while* iteration < *itrmax do*

7: Increment iteration by 1

8: Compute the new velocity of each particle as,
velocity(new)=w*velocity(old)+ c1*(pbest- position(old))+c2*(gbest-position(old))

9: Update the position of each particle by,
position(new)=position(old)+velocity(new)

10: Compute fitness of each particle with its updated position using objection function in Equation (1)

11: Update *the pbest* of each particle and the *gbest* of all particles

12: end while

Return best chromosome

Algorithm 3. Parameter optimization of member functions using DE

Input: Population size (*n*), maximum number of
Iterations (*itrmax*), crossover probability (*CR*).

Output: Membership functions' parameters.

1: Generate initial target vectors (*nx8*)

2: Set iteration=0

3: Compute the fitness of the target vectors using objective function in Equation (1) and find the best target vector *pbest*

4: *while* iteration < *itrmax do*

5: Increment iteration by 1

6: Generate F by the Quantile function of the Cauchy distribution

7: *for i=1:n*

8: Generate donor vector by,
donor vector(i)=target vector(i)+ F(pbest-target vector(i)+target vector(r1)-target vector(r2)), such that r1≠r2≠i

9: Perform recombination between target vector(*i*) and donor vector(*i*) with *CR* to generate trial vector(*i*)

10: Compute the fitness of the trial vector(*i*) using objective function in Equation (1)

11: Replace the target vector(*i*) by trial vector(*i*), if trial vector is fitter
end for

12: Update target vectors and *pbest* for the next iteration

13: *end while*

Return best chromosome

3.2. Fuzzy Rule Base

To illustrate the proposed method, a single input-single output system is considered with a set of rules (R1-R3) as given below.

R1: If input is low, then output is low.
R2: If input is medium, then output is medium.
R3: If input is high, then output is high.

3.3. Fuzzy Inference Engine

The Fuzzy Inference Engine (FIS) infers the three rules in the rule base with different firing strengths of [0, 1]. The final fuzzy output of the FIS is obtained by aggregating the outputs of all three rules using the fuzzy MAX operator.

3.4. Defuzzification

There are different defuzzification techniques [31], each with some advantages and disadvantages. The proposed model uses one of the most popular defuzzification techniques, Mean of Maxima (MOM) [32], to find the final crisp output.

4. Simulation Results

The performance of the proposed fuzzy system is validated on four nonlinear functions, viz., a cube function, a square function, a square root function, and an exponential function, as given in Equations (3) to (6) respectively.

$$y = x^3 \quad \text{for} \quad 0 \le x \le 2 \tag{3}$$

$$y = x^2 \quad \text{for} \quad 0 \le x \le 3 \tag{4}$$

$$y = \sqrt{x} \quad \text{for} \quad 0 \le x \le 9 \tag{5}$$

$$y = e^x \quad \text{for} \quad 0 \le x \le 3 \tag{6}$$

Where x and y are the notions of input and output, respectively, to quantitatively measure and compare the outputs of the suggested fuzzy model and the conventional fuzzy model, two error metrics, viz., average sum squared error ((here denoted by *ASSE*) and average error (here denoted by *AE*) are used. The expressions of these two error indices are given by Equations (7) and (8).

$$ASSE = \frac{1}{r}\sum_{k=1}^{r}(e(k) - p(k))^2 \tag{7}$$

$$AE = \frac{1}{r}\sum_{k=1}^{r}|e(k) - p(k)| \tag{8}$$

Here, *e(k)* and *p(k)* represent the exact output and predicted output of the *k*-th test point, and *r is* the size of the test points. For unbiased comparison, the shape of the membership functions for the conventional fuzzy system is also considered Gaussian, with the low subset and high subset as the right-sided Gaussian and left-sided Gaussian, respectively. The mean of the subset middle of the conventional fuzzy system is set at the midpoint of the input variable for the antecedent and the midpoint of the output variable for the consequent. The standard deviations of the membership functions for the conventional fuzzy system are all calculated as 1/6th of the respective variable.

In this paper, the conventional fuzzy model is denoted by CFM and the Proposed Fuzzy Models (PFMs), whose membership functions are generated by GA, PSO, and DE, are denoted by PFM-GA, PFM-PSO, and PFM-DE, respectively. The performances of the fuzzy models (CFM, PFM-GA, PFM-PSO, and PFM-DE) are tested on 11 equally spaced data points of the four nonlinear functions considered here. The actual and predicted outputs for the test data points of the cube function, square function, square root function, and exponential function are depicted in Figures 3, 4, 5, and 6, respectively.

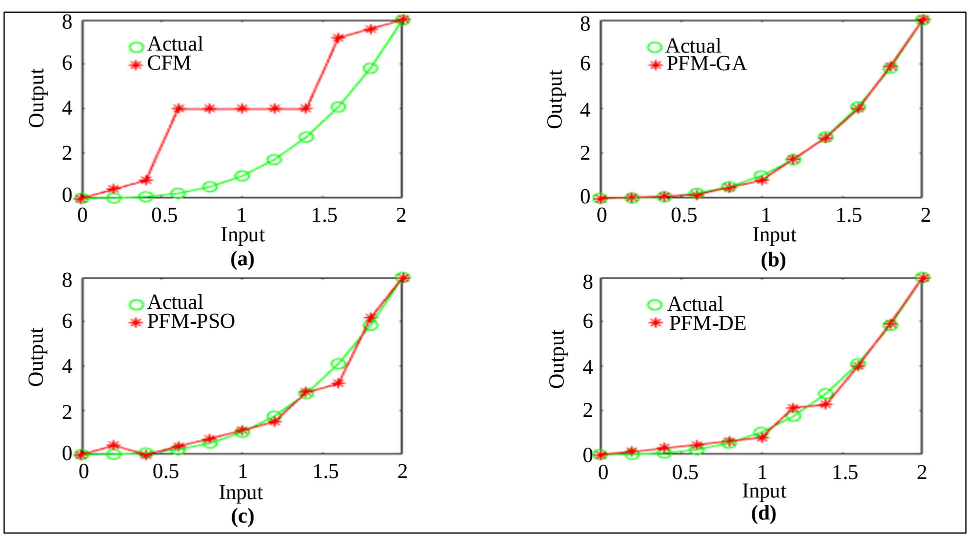

Fig. 3 Actual (o) and predicted (*) output for the cube function as estimated by (a) CFM (b) PFM-GA (c) PFM-PSO (d) PFM-DE

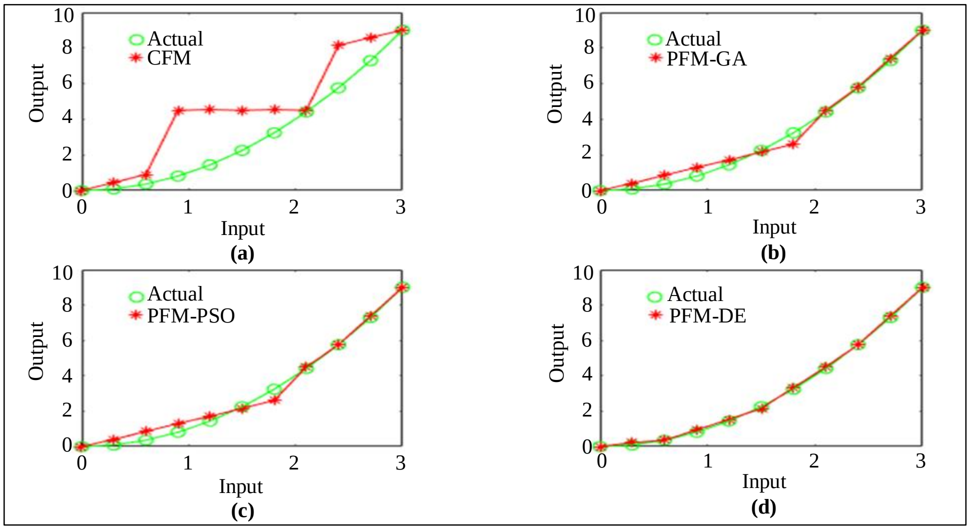

Fig. 4 Actual (o) and predicted (*) output for the square function as estimated by (a) CFM, (b) PFM-GA, (c) PFM-PSO, and (d) PFM-DE.

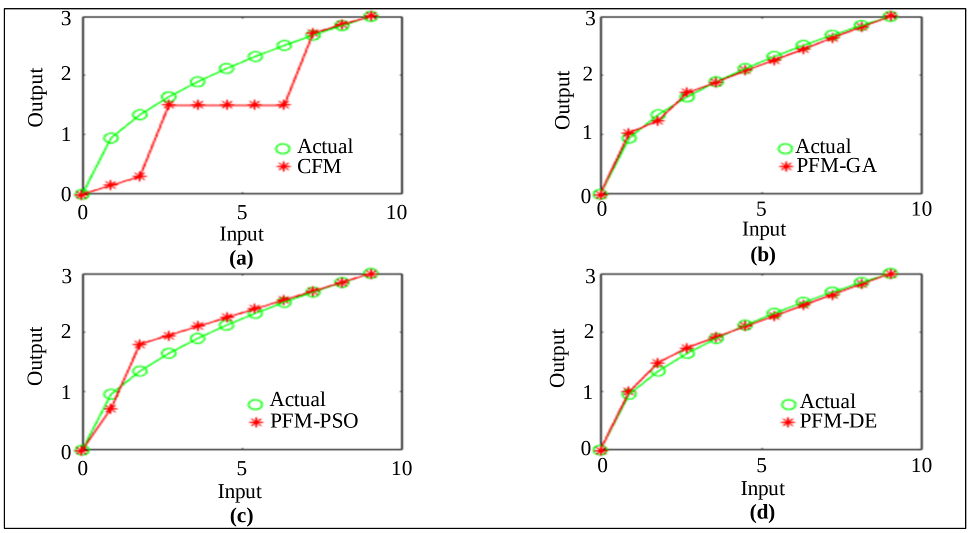

Fig. 5 Actual (o) and predicted (*) output for the square root function as estimated by (a) CFM, (b) PFM-GA, (c) PFM-PSO, and (d) PFM-DE.

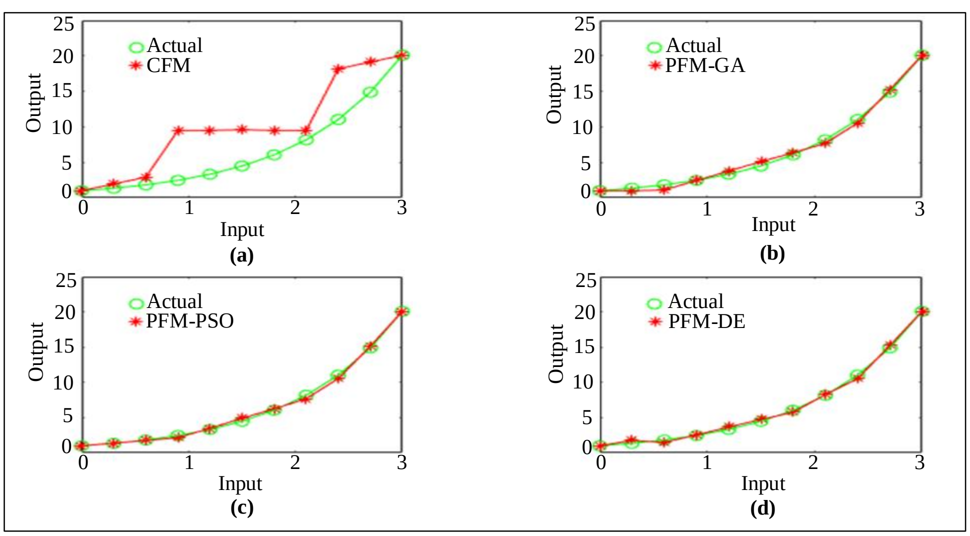

Fig. 6 Actual (o) and predicted (*) output for the exponential function as estimated by (a) CFM (b) PFM-GA (c) PFM-PSO (d) PFM-DE

Table 1. Comparative performances of the fuzzy models with respect to ASSE

Test Functions	Fuzzy Models			
	CFM	PFM-GA	PFM-PSO	PFM-DE
Cube	5.061847	0.005818	0.109382	0.054487
Square	3.440291	0.099593	0.098489	0.006995
Square root	0.362491	0.002987	0.038995	0.003597
Exponential	18.006377	0.173327	0.080090	0.082414

Table 2. Comparative performances of the fuzzy models with respect to AE

Test Functions	Fuzzy Models			
	CFM	PFM-GA	PFM-PSO	PFM-DE
Cube	1.800000	0.052364	0.219636	0.178182
Square	1.366364	0.233182	0.225000	0.065455
Square root	0.444238	0.044234	0.134281	0.044415
Exponential	3.289064	0.343060	0.218723	0.236512

The errors in predicting the four non-linear functions were noted, and the values of the error indices are presented in Tables 1 and 2 to compare the performances of the suggested fuzzy model and the conventional fuzzy model quantitatively.

5. Result Discussion

From Figure 3(a), it is seen that the Conventional Fuzzy Model (CFM) has predicted the correct output of the cube function at very few points, rising to the very high values of ASSE (approximately 5.062) and AE (approximately 1.80), as are found from Tables 1 & 2. From Figures 3(b) - 3(d), it is seen that all the proposed models (PFM-GA, PFM-PSO, and PFM-DE) are outperforming the conventional fuzzy model. Among the proposed models, PFM-PSO & PFM-DE are predicting the cube function at some points accurately, whereas, at some other points, there exist minor deviations from the actual outputs.

From Tables 1 and 2, it is seen that the average sum squared error for PFM-PSO & PFM-DE are 0.109 and 0.054, respectively, and the average error for PFM-PSO & PFM-DE are 0.220 and 0.178, respectively. The best performance in the prediction of the cube function is obtained by PFM-GA. From Figure 3(b), it is noticed that except at a very few points, PFM-GA has predicted the output accurately. From Tables 1 and 2, it appears that the values of ASSE & AE in predicting the cube functions at the test points by PFM-GA are 0.006 and 0.052, respectively. Similar to the cube function, the square function has not been predicted well by the conventional fuzzy model, as seen in Figure 4(a), and it encompasses very large errors, as found in the result tables (Tables 1 and 2). From Figures 4(b) and 4(c), it is seen that the performance of PFM-GA and PFM-PSO are very similar, and they predict the square function accurately at some points and deviate from the actual outputs at some other points.

The square function has been quite satisfactorily predicted by PFM-DE, as seen in Figure 4(d). It is further supported by the values of the error indices, as shown in Tables 1 and 2. PFM-DE, with values of 0.007 and 0.0655 for ASSE and AE, respectively, outperforms the other three methods in predicting the square function. Figure 5(a) indicates that for the square root function, the performance of the conventional fuzzy model is not appreciable, and it is imperative from Tables 1 and 2, too.

From Figures 5(b)-5(d), it is noticed that PFA-GA and PFM-DE are making sufficiently good predictions, whereas PFM-PSO, although performing better than the conventional fuzzy model, at some points, its predicted outputs are much deviated from the actual output values. From the tables of results, it is seen that among the four models, the square root function has been best predicted by the PFM-GA. The exponential function, as seen from Figures 6(b)–6(d), at the test points is predicted very well by all three proposed models.

In terms of errors, this function has been best predicted by PFM-PSO, as found in Tables 1 and 2.

6. Conclusion

This paper proposes a novel optimization-based technique for designing a fuzzy model with Gaussian-shaped membership functions. The simulation results demonstrate that the proposed model outperforms the traditional fuzzy model in approximating the four nonlinear functions considered here. It is expected that the suggested fuzzy modeling technique will be adopted in various domains of fuzzy-based system design to generate membership functions from data. The suggested technique has been applied to model a Mamdani-type fuzzy system. Efforts may be put into designing a Sugeno-type fuzzy system utilizing the proposed method. Further research may also be conducted to achieve better results using multi-objective optimization algorithms instead of single-objective optimization algorithms in the design process.

Acknowledgments

The authors would like to thank their institution for providing a good research ambience that has helped them to conduct work related to this article.

References

[1] L.A. Zadeh et al., "Fuzzy Sets," *Information and* Control, vol. 8, no. 3, pp. 338-353, 1965. [CrossRef] [Google Scholar] [Publisher Link]

[2] Lotfi A Zadeh, Rafik A. Aliev, *Fuzzy Logic Theory and Applications: Part I and Part II*, World Scientific, Singapore, 2018. [CrossRef] [Google Scholar] [Publisher Link]

[3] Elena Vlamou, and Basil Papadopoulos, "Fuzzy Logic Systems and Medical Applications," *AIMS Neuroscience*, vol. 6, no. 4, pp. 266-272, 2019. [CrossRef] [Google Scholar] [Publisher Link]

[4] Hangyao Wu, and Zeshui XU, "Fuzzy Logic in Decision Support: Methods, Applications and Future Trends," *International Journal of Computers Communications & Control*, vol. 16, no. 1, pp. 1-27, 2021. [CrossRef] [Google Scholar] [Publisher Link]

[5] E.H. Mamdani, and S. Assilian, "An Experiment in Linguistic Synthesis with a Fuzzy Logic Controller," *International Journal of Man-Machine Studies*, vol. 7, no.1, pp. 1-13, 1975. [CrossRef] [Google Scholar] [Publisher Link]

[6] Tomohiro Takagi, and Michio Sugeno, "Fuzzy Identification of Systems and Its Applications to Modeling and Control," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 15, no.1, pp. 116-132, 1985. [CrossRef] [Google Scholar] [Publisher Link]

[7] Dimitar P. Filev, and Ronald R. Yager, *Essential of Fuzzy Modeling and Control*, John Wiley & Sons Inc., Singapore, 1994. [Google Scholar]

[8] Ajit K. Mandal, *Introduction to Control Engineering Modeling, Analysis and Design*, New Age International Publisher, India, 2006. [Google Scholar]

[9] Swarup Medasani, Jaeseok Kim, and Raghu Krishnapuram, "An Overview of Membership Function Generation Techniques for Pattern Recognition," *International Journal of Approximate Reasoning*, vol. 19, no. 3-4, pp. 391-417, 1998. [CrossRef] [Google Scholar] [Publisher Link]

[10] I. Burhan Türkşen, "Review of Fuzzy System Models with an Emphasis on Fuzzy Functions," *Transactions of the Institute of Measurement and Control*, vol. 31, no. 1, pp. 7-31, 2009. [CrossRef] [Google Scholar] [Publisher Link]

[11] Mehdi Zangeneh, Ebrahim Aghajari, and Mehdi Forouzanfar, "A Survey: Fuzzify Parameters and Membership Function in Electrical Applications," *International Journal of Dynamics & Control*, vol. 8, pp. 1040-1051, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[12] T.W. Liao, Aivars K. Celmins, and Robert J. Hammell II, "A Fuzzy C-Means Variant for the Generation of Fuzzy Term Sets," *Fuzzy Sets and Systems*, vol. 135, no. 2, pp. 241-257, 2003. [CrossRef] [Google Scholar] [Publisher Link]

[13] Muhammad Hamza Azam et al., "Fuzzy Type-1 Triangular Membership Function Approximation Using Fuzzy C-Means," *2020 International Conference on Computational Intelligence (ICCI)*, Bandar Seri Iskandar, Malaysia, pp. 115-120, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[14] Chun-Hao Chen et al., "Cluster-Based Membership Function Acquisition Approaches for Mining Fuzzy Temporal Association Rules," *IEEE Access*, vol. 8, pp. 123996-124006, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[15] Chih-Chung Yang, and N.K. Bose, "Generating Fuzzy Membership Function with Self-Organizing Feature Map," *Pattern Recognition Letters*, vol. 27, no. 5, pp. 356-365, 2006. [CrossRef] [Google Scholar] [Publisher Link]

[16] Imen Derbel, Narjes Hachani, and Habib Ounelli, "Membership Functions Generation Based on Density Function," *2008 International Conference on Computational Intelligence and Security*, Suzhou, China, pp. 96-101, 2008. [CrossRef] [Google Scholar] [Publisher Link]

[17] Dumidu Wijayasekara, and Milos Manic, "Data-Driven Fuzzy Membership Function Generation for Increased Understandability," *2014 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)*, Beijing, China, pp. 133-140, 2014. [CrossRef] [Google Scholar] [Publisher Link]

[18] Hossein Pazhoumand-Dar, Chiou-Peng Lam, and Martin Masek, "Automatic Generation of Fuzzy Membership Functions Using Adaptive Mean-Shift and Robust Statistics," *Proceedings of 8th International Conference on Agents and Artificial Intelligence*, pp. 160-171, 2016. [CrossRef] [Google Scholar] [Publisher Link]

[19] Charles L. Karr, "Design of a Cart-Pole Balancing Fuzzy Logic Controller Using a Genetic Algorithm," *Proceedings of the SPIE*, vol. 1468, 1991. [CrossRef] [Google Scholar] [Publisher Link]

[20] F. Herrera, M. Lozano, and J.L. Verdegay, "Tuning Fuzzy Logic Controllers by Genetic Algorithms," *International Journal of Approximate Reasoning*, vol. 12, no. 3-4, pp. 299-315, 1995. [CrossRef] [Google Scholar] [Publisher Link]

[21] Ahmet Arslan, and Mehmet Kaya, "Determination of Fuzzy Logic Membership Functions Using Genetic Algorithms," *Fuzzy Sets and Systems*, vol. 118, no. 2, pp. 297-306, 2001. [CrossRef] [Google Scholar] [Publisher Link]

[22] Huai-Xiang Zhang, Feng Wang, and Bo Zhang, "Genetic Optimization of Fuzzy Membership Functions," *2009 International Conference on Wavelet Analysis and Pattern Recognition*, Baoding, pp. 465-470, 2009. [CrossRef] [Google Scholar] [Publisher Link]

[23] B. Safaee, and S.K. Kamaleddin Mousavi Mashhadi, "Optimization of Fuzzy Membership Functions via PSO and GA with Application to Quadrotor," *Journal of AI and Data Mining*, vol. 5, no. 1, pp. 1-10, 2017. [CrossRef] [Google Scholar] [Publisher Link]

[24] Amit Konar, *Computational Intelligence: Principles, Techniques and Applications*, 1st ed., Springer, Berlin Heidelberg, 2005. [CrossRef] [Google Scholar] [Publisher Link]

[25] S. Roy, and Udit Chakraborty, *Introduction to Soft Computing, Neuro-Fuzzy and Genetic Algorithms*, Pearson, India, 2013. [Google Scholar]

[26] Sourabh Katoch, Sumit Singh Chauhan, and Vijay Kumar, "A Review on Genetic Algorithm: Past, Present, and Future," *Multimedia Tools and Applications*, vol. 80, pp. 8091-8126, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[27] Dongshu Wang, Dapei Tan, and Lei Liu, "Particle Swarm Optimization Algorithm: An Overview," *Soft Computing*, vol. 22, pp. 387-408, 2018. [CrossRef] [Google Scholar] [Publisher Link]

[28] Rainer Storn, and Kenneth Price, "Differential Evolution - A Simple and Efficient Heuristic for Global Optimization over Continuous Spaces," *Journal of Global Optimization*, vol. 11, no. 4, pp. 341-359, 1997. [CrossRef] [Google Scholar] [Publisher Link]

[29] Swagatam Das, and Ponnuthurai Nagaratnam Suganthan, "Differential Evolution: A Survey of the State-of-the-Art," *IEEE Transactions on Evolutionary Computation*, vol. 15, no. 1, pp. 4-31, 2011. [CrossRef] [Google Scholar] [Publisher Link]

[30] Manolis Georgioudakis, and Vagelis Plevris, "A Comparative Study of Differential Evolution Variants in Constrained Structural Optimization," *Frontiers in Built Environment*, vol. 6, 2020. [CrossRef] [Google Scholar] [Publisher Link]

[31] Shounak Roychowdhury, and Witold Pedrycz, "A Survey of Defuzzification Strategies," *International Journal of intelligent systems*, vol. 16, no. 6, pp. 679-695, 2001. [CrossRef] [Google Scholar] [Publisher Link]

[32] Aurélie Talon, and Corinne Curt, "Selection of Appropriate Defuzzification Methods: Application to the Assessment of Dam Performance," *Expert Systems with Applications*, vol. 70, pp. 160-174, 2017. [CrossRef] [Google Scholar] [Publisher Link]