Towards More Flexible Fuzzy Membership Functions: Learning from Data

Fakhri Abbasov
School of Computer Science
University of Nottingham
Nottingham, UK
psyfa3@nottingham.ac.uk

Chao Chen
School of Computer Science
University of Nottingham
Nottingham, UK
chao.chen@nottingham.ac.uk

Jonathan M. Garibaldi
School of Computer Science
University of Nottingham
Nottingham, UK and Ningbo, China
jon.garibaldi@nottingham.edu.cn

***Abstract*—Fuzzy systems are widely recognised for their ability to model uncertainty and linguistic knowledge, but their effectiveness often depends on the choice of membership functions. Traditional approaches have relied on membership functions with predefined, convex shapes to facilitate interpretability and computational efficiency. However, such constraints can limit the expressive power of fuzzy systems. This paper advocates for a more flexible and data-driven approach to fuzzy membership function design, using B-splines as a powerful example of adaptable, learnable shapes. Unlike previous approaches that impose boundary conditions, the learned B-splines are allowed to assume any shape required by the application domain. To optimise these flexible membership functions, an Adaptive Network-based Fuzzy Inference System (ANFIS) is integrated with gradient-based techniques enabled by PyTorch's automatic differentiation engine. This integration eliminates the need for manually derived gradients for each B-spline control point, thereby enhancing both the efficiency and reliability of the training process. Experimental evaluations on real-world datasets demonstrate that this flexible, data-driven approach captures complex, non-convex relationships effectively, leading to improved predictive performance and robustness.**

***Index Terms*—Flexible Membership Functions, B-splines, ANFIS, Automatic Differentiation**

I. INTRODUCTION

Since their introduction by Zadeh in 1965 [1], fuzzy sets have provided a powerful framework for modelling uncertainty and imprecision inherent to human reasoning. Fuzzy set theory enabled the development of fuzzy logic controllers and systems, which have been widely used in various control and modelling applications [2, 3, 4]. At the core of any fuzzy system lies the membership function (MF), which defines how input values are mapped to degrees of membership in a fuzzy set. The shape and properties of these MFs are crucial, as they directly influence the system's interpretability, reasoning capabilities, and overall performance. Traditional fuzzy systems have relied on commonly used MF shapes such as triangular, trapezoidal, and Gaussian that are interpretable and computationally efficient. These conventional MFs are typically convex and normal. However, such predefined MFs may not accurately capture the complexity of real-world data relationships in some problems [5, 6]. For example, the "drinkability" of milk may peak at both cold and hot temperatures but decrease in between, leading to a non-convex function. Despite their potential, the adoption of non-convex MFs has been limited by the lack of general frameworks for their design, parameterisation, and integration into fuzzy systems.

There is also the challenge of selecting the appropriate MFs for a specific application. As highlighted by Wang et al. [7], the choice of MFs is often based on subjective perceptions rather than on data or other objective factors. The lack of a unified form of fuzzy MFs reduces the adaptability of fuzzy systems. Consequently, there is a growing need for MF frameworks that are capable of adapting to the unique characteristics of the application domain.

Wang et al. [7] demonstrated that B-splines can be used to construct MFs, effectively creating smooth, locally adjustable curves. B-splines are piecewise polynomial functions that provide a flexible way to model smooth curves by defining a set of control points and blending functions. Unlike traditional parametric MFs, B-splines offer local control, meaning that adjustments to one part of the function do not affect the entire curve. In Wang's study, B-spline MFs are either derived from empirical data or trained in a fuzzy neural network using predefined shapes and rules provided by experts. Although a gradient descent method was used for training, it tuned the output values of the MFs rather than adjusting the control points directly, and the experiments were limited to linear splines. Additionally, boundary constraints were imposed: each MF was required to have zero slope at both ends (S-shaped conditions) and to equal 0 at one end and 1 at the other, which enforces normality.

Optimising MFs remains a significant challenge in fuzzy systems. Even when an initial rule base and a set of MFs are determined, fine-tuning these elements is necessary for optimal performance. Early methods relied on fixed, parametric functions and heuristics to make adjustments, which could fail to capture complex patterns in real-world data. Researchers have also explored evolutionary algorithms and metaheuristics such as Genetic Algorithms [8], Particle Swarm Optimisation [9], and Ant Colony Optimisation [10] to automate the tuning process. While these methods can be effective, they are computationally expensive and may not be applicable to largescale problems. Alternatively, adaptive systems such as the Adaptive Network-Based Fuzzy Inference System (ANFIS) have been developed, which leverage the learning capabilities of neural networks to optimise the fuzzy system parameters [11, 12]. ANFIS benefits from automatically tuning both MFs and rule weights through gradient-based methods, enabling it to capture complex patterns. However, this approach requires the derivation of gradients for the implicit parameters, which, if done manually, can be time consuming and prone to error.

Chen et al. [13] demonstrated the integration of automatic differentiation into the FuzzyR library for the R programming language [14]. Their work highlights how autograd can be used to simplify the computation of derivatives for MF parameters, reducing the complexity of manual gradient derivations and improving efficiency in fuzzy system optimisation. This integration provides a foundation for extending automatic differentiation to more flexible fuzzy models.

In this paper, the potential of using B-spline MFs as a flexible and data-driven alternative to traditional functions is explored. B-splines are integrated with an ANFIS framework, and autograd is leveraged to efficiently compute the required gradients. This approach is evaluated on multiple real-world datasets, and its performance is compared with conventional methods.

II. Non-Convex Membership Functions

A. Definitions

- **Normal:** A fuzzy set A is normal if there exists at least one $x' \in X$ such that $\mu_A(x') = 1$.
- **Convex:** A fuzzy set A is convex if, for every α-cut A_α, any two points $r, s \in A_\alpha$ satisfy $\lambda r + (1-\lambda)s \in A_\alpha$ for all $\lambda \in [0, 1]$.

B. Non-Convex and Sub-Normal Membership Functions

Traditional fuzzy MFs in many applications are often required to be normal and convex. However, such constraints can be overly restrictive when modelling real-world phenomena. Research has shown that relaxing these assumptions can enhance the expressive power of fuzzy systems. In particular, it was pointed out that enforcing normality and convexity in MFs is largely an empirical convention in fuzzy control rather than a theoretical necessity [5, 6]. Their findings suggest that removing these constraints could provide greater flexibility in fuzzy modelling, potentially leading to improved performance in practical applications.

More specifically, Garibaldi et al. [6] identified three cases in which relaxing traditional constraints is beneficial:

- **Elementary Non-Convex Sets:** Consider a scenario like forming a mountain rescue team. The suitability of different team sizes might be influenced by several factors. For instance, the need for an odd number of members to avoid voting ties, the potential for interpersonal issues in very small teams, and the complications arising from too many members. These competing considerations can lead to a discrete fuzzy set with multiple peaks, reflecting the complex trade-offs in determining an "ideal" team size.

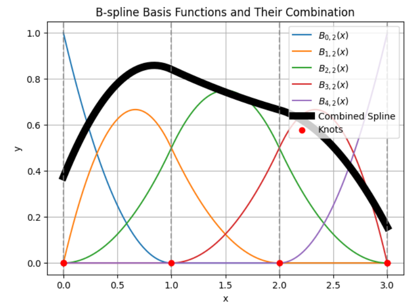


Fig. 1: Illustration of a B-spline. The knot vector partitions the input domain, while the individual basis functions (depicted in different colours) combine to generate the overall B-spline curve.

- **Time-Related Non-Convex Sets:** Some concepts naturally vary over time. For instance, energy demand or the notion of mealtime can be modelled with fuzzy sets that show multiple sub-peaks corresponding to breakfast, lunch, and dinner. Such MFs may not drop to zero between these peaks, resulting in sub-normal shapes that better represent the continuous and overlapping nature of these temporal phenomena.
- **Consequent Non-Convex Sets:** In rule-based fuzzy systems (such as those using Mamdani inference), the aggregation of several fuzzy rules can produce output MFs that are non-convex. While these functions are often defuzzified to yield precise numerical outputs in control applications, their inherently non-convex shapes can be directly useful for modelling human decision-making or for further chaining in inference processes.

C. B-Spline Functions as Membership Functions

B-splines, or basis splines, are piecewise polynomial functions that offer smooth transitions and local control.

A B-spline is defined by the following:

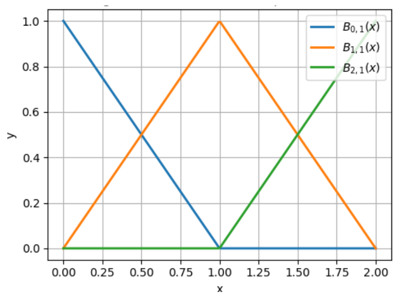
- Degree (p): The polynomial order (e.g. $p = 3$ for a cubic spline).
- Control Points: The coefficients that shape the MF.
- Knot Vector: A sequence dividing the input domain. For n control points and degree p, the knot vector has $n+p+1$ entries.

In this work, B-spline basis functions are computed using the well-established Cox-de Boor algorithm [15], which provides a robust recursive formulation. Specifically, the basis functions are defined as follows:

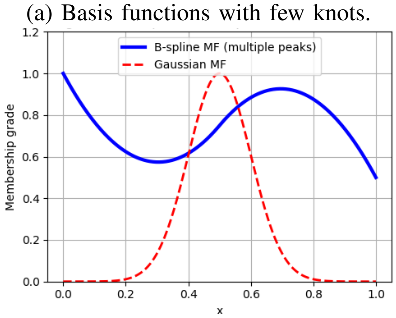
$$B_{k,0}(x) = \begin{cases} 1, & \text{if } t_k \leq x < t_{k+1}, \\ 0, & \text{otherwise,} \end{cases}$$

and for $p \geq 1$,

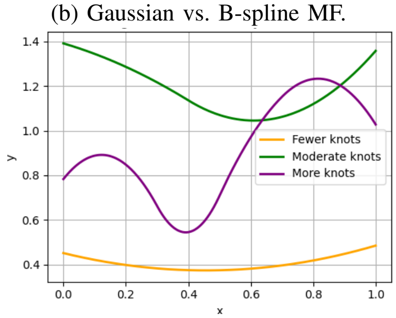
$$B_{k,p}(x) = \frac{x - t_k}{t_{k+p} - t_k} B_{k,p-1}(x) + \frac{t_{k+p+1} - x}{t_{k+p+1} - t_{k+1}} B_{k+1,p-1}(x).$$



(a) Basis functions with few knots.



(b) Gaussian vs. B-spline MF.



(c) Detail vs. Interpretability.

Fig. 2: Illustrations of the adaptability of B-spline MFs. (a) Basis functions with a few knots yield a smooth curve. (b) A comparison shows the increased flexibility of a learned B-spline MF relative to a Gaussian MF. (c) Increasing the knot density enhances fine-grained detail but introduces complexity, potentially reducing interpretability.

The basis functions are the building blocks of B-splines. To form a complete B-spline curve, multiple basis functions are linearly combined using control point coefficients (c_k):

$$S(x) = \sum_{k=0}^{n} c_k B_{k,p}(x),$$

where c_k represents the influence of each control point. This combination enables the construction of complex, smooth curves or MFs.

B-splines are particularly well-suited to represent non-convex and sub-normal MFs because they can naturally form multiple peaks based on how knots and control coefficients are learned, and there is no requirement for the function to attain a maximum membership value of 1.

Figure 1 presents the construction of a B-spline curve using its basis functions. Figure 2a demonstrates a set of B-spline basis functions with 5 knots and a degree of 1 that result in relatively broad triangular-shaped pieces. Figure 2b highlights the adaptability of B-splines by comparing a single, multi-peaked B-spline MF to a classic Gaussian MF, demonstrating how B-splines can represent shapes that deviate significantly from convex shapes. Finally, Figure 2c explores the trade-off between detail and interpretability by comparing B-splines with varying knot densities.

III. Experiments and Results

This section evaluates the interpretability and representational power of B-spline MFs, which are optimised using an adapted ANFIS framework with automatic differentiation. An objective is to assess whether B-spline MFs provide meaningful and interpretable results compared to traditional MFs. Furthermore, the experiments will compare the performance of B-spline-enhanced ANFIS with other models, including those using conventional MFs and a standalone neural network, to demonstrate their effectiveness in capturing complex relationships.

A. Datasets

Three regression datasets were used in the evaluation:

1) **Laser**: A univariate time series of 1000 points recorded from a Far-Infrared Laser operating in a chaotic state [16]. Each instance was constructed using four consecutive time steps as inputs and the fifth as the target, resulting in 993 instances.
2) **Ele1 (ele1-2-495)**: A dataset with 495 examples, each containing two input variables and one output [17]. It is derived from a real-world application that estimates the total length of low-voltage electrical lines in rural towns.
3) **Ele2 (ele2-4-1056)**: A dataset of 1056 examples used for estimating medium-voltage maintenance costs [17]. It includes four input variables and one output representing the cost estimate.

The evaluation used 5-fold cross-validation, with each fold serving as the test set once while the others were used for training. The process was repeated 10 times to ensure statistical robustness.

B. Baseline Model Details.

Two baseline models are used as benchmarks:

- **ANFIS (Gaussian)**: Two Gaussian MFs per input $\texttt{n\_mfs} = 2$) are used. The mean and variance parameters are optimised using gradient descent and least-squares estimation, following the approach in [18].
- **MLP**: A fully connected neural network with two hidden layers (64 neurons each), ReLU activations, and an output layer for regression. The model is trained using the Adam Optimiser (learning rate: 0.01, batch size: 64), with hyperparameters aligned to those in the fuzzy models for comparability.

C. B-Spline based ANFIS and Tunable Parameters

The B-spline-enhanced ANFIS model, which incorporates flexible and tunable membership functions, is designed for comparison with baseline models. In this model, each MF, $\mu_j(x)$, is constructed as a weighted combination of B-spline basis functions. Let $\alpha_{j,k}^{\text{raw}}$ be the raw, unconstrained trainable parameters associated with the j-th MF. The coefficients $\beta_{j,k}$ used in the spline construction are obtained by applying the sigmoid function for stability:

$$\beta_{j,k} = \sigma(\alpha_{j,k}^{\text{raw}}), \quad \text{where} \quad \sigma(z) = \frac{1}{1+e^{-z}}.$$

The membership grade $\mu_j(x)$ is then calculated as:

$$\mu_j(x) = \text{clamp}\left(\sum_k \beta_{j,k} B_{k,p}(x),\ 0,\ 1\right)$$

Here, $B_{k,p}(x)$ are the B-spline basis functions of degree p defined over a fixed knot vector, and the summation includes all basis functions $B_{k,p}$ relevant for the input x. The $\text{clamp}(z, 0, 1)$ function ensures the output is strictly bounded between 0 and 1, mapping values below 0 to 0 and values above 1 to 1. This final clamping guarantees that the output follows the definition of a fuzzy membership degree.

The knot vector for each B-spline MF is initialised to uniformly partition the input domain, and the coefficients $\alpha_{j,k}^{\text{raw}}$ are randomly initialised. The first and last knots are chosen by taking the minimum and maximum of the input data range and extending them slightly by a δ. This small margin ensures that the spline can properly interpolate values at the boundaries without excessively constraining its shape near the ends of the input domain. Additionally, a standard scaling procedure is applied to each dataset to normalise the input features to zero mean and unit variance, which stabilises gradient-based updates during training and promotes efficient convergence.

A two-phase training strategy is adopted to optimise the model parameters. In Phase 1, all parameters, including B-spline coefficients and consequent parameters, are updated using the Adam optimiser for robust gradient-based learning. In Phase 2, while the MF parameters continue to be updated via gradient descent, the consequent parameters are recalculated using least-squares estimation to improve convergence efficiency.

To mitigate overfitting, the L2 regularisation is applied to both MF and the subsequent parameters. This technique penalises large parameter values, promoting simpler and more generalisable patterns while stabilising the training process. Such regularisation is crucial given the flexibility of B-spline MFs, which can otherwise risk overfitting.

D. Hyperparameters

The B-spline-enhanced ANFIS was configured with the following hyperparameters:

- `n_mfs` = 2: The number of MFs assigned to each input variable.
- `spline_degree` = 3: The degree of the B-spline functions.
- `n_basis` = 6: The number of basis functions used to construct each B-spline, affecting the level of detail captured by the MF.
- `lambda_reg` = 0.001: The regularisation coefficient applied during the hybrid learning phase to the consequent parameters.
- `epochs` = 300: The total number of training epochs during which gradient-based optimisation is performed.
- `hybrid_epochs` = 50: The number of epochs dedicated to the hybrid optimisation phase, where consequent parameters are updated using a least-squares method.
- `learning_rate` = 0.1: The initial learning rate for the optimiser, which controls the step size during gradient descent.
- `batch_size` = 64: The number of training examples used in each mini-batch during the optimisation process.
- `weight_decay` = 0.001: The regularisation parameter used with the Adam optimiser to penalise large weights in all parameters *except* those updated via the least-squares method.

E. Quantitative Results

Table I reports the Mean Squared Error (MSE) for each method. The results indicate that the B-spline-enhanced ANFIS consistently outperforms both the Gaussian ANFIS and the MLP baseline, achieving lower MSE values and reduced variability across datasets.

TABLE I: MSE Comparison (mean and standard deviation) Across Datasets

Method	**Mean MSE**	**Std. Dev.**
Laser Dataset		
B-Spline ANFIS	35.16	15.57
Gaussian ANFIS	62.26	34.46
MLP	99.74	71.55
Ele1 Dataset		
B-Spline ANFIS	413073	66628
Gaussian ANFIS	526338	198389
MLP	487180	206171
Ele2 Dataset		
B-Spline ANFIS	8815	1286
Gaussian ANFIS	13730	2241
MLP	27109	2270

F. Discussion on Learned MF Shapes

Figures 3 and 4 illustrate the differences between B-spline-based and Gaussian MFs on the Ele1 dataset. The learned B-spline MFs display notable advantages in terms of flexibility and adaptability to the underlying data distribution. These characteristics are particularly evident when examining the following aspects:

a) Flexibility to Represent Complex Shapes: The B-spline MFs (Figure 3) show non-convex and multi-peaked shapes, with the ability to adapt to fine-grained variations in the data. For example, the first input feature exhibits two distinct peaks, which likely correspond to different population

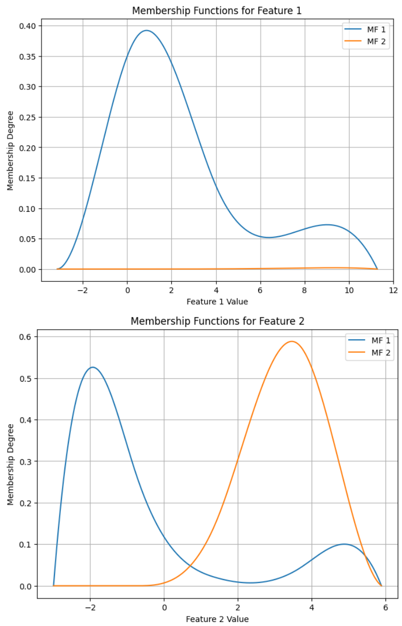


Fig. 3: Learned B-spline MFs for the two Ele1 input features. Note the presence of multiple local maxima and sub-normal peaks in each MF. In addition, one of the MFs remains near zero across the entire domain, indicating that the data did not meaningfully leverage that fuzzy set.

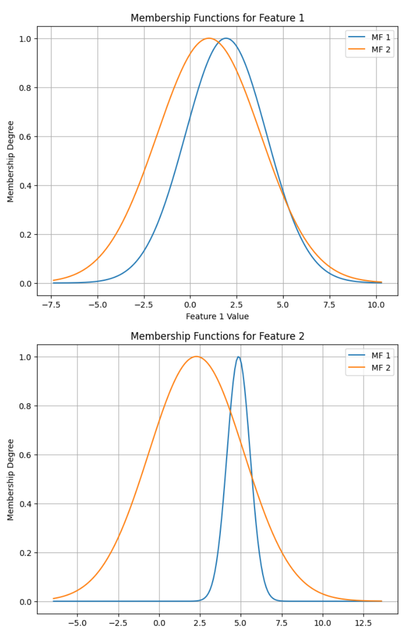


Fig. 4: Learned Gaussian MFs for the same Ele1 input features. As expected, these functions are normal, unimodal, and relatively symmetric. Note, however, that they are strongly overlapping, suggesting that the second MF does not capture a distinctly different input region.

ranges where the low-voltage line length sharply varies based on the town's geographic layout. Such complex shapes cannot

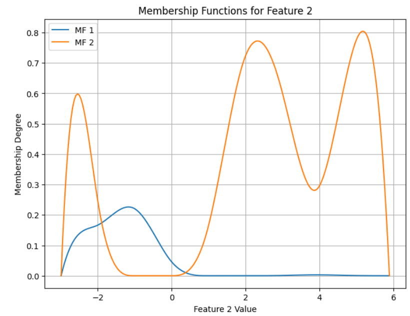


Fig. 5: Learned B-spline membership function for feature 2 of the Ele1 dataset using 10 basis functions. The increased number of basis functions yields a more complex shape with multiple peaks.

be captured by the unimodal Gaussian MFs (Figure 4), which are restricted by their fixed and symmetric form. This demonstrates the ability of B-splines to represent more nuanced relationships that are often observed in real-world datasets.

b) Sub-normality as a Reflection of Data Distribution: Some B-spline MFs exhibit sub-normal behaviour (membership values below 1), suggesting that the data do not strongly support a single "degree of truth" for certain input regions. This flexibility allows B-spline MFs to assign moderate membership values across broader areas where the data lack clear separation. In contrast, Gaussian MFs are always normalised, potentially leading to misrepresentation in such ambiguous regions. The ability to naturally represent sub-normality aligns well with fuzzy logic's core principles, enabling more realistic modelling of uncertainty.

c) Pruning Unnecessary Membership Functions: One noteworthy observation is that certain B-spline MFs remain near zero across the entire domain. This indicates that the model effectively prunes unnecessary MFs when the data suggests that they are not needed, thereby optimising the use of fuzzy sets. On the other hand, Gaussian MFs tend to overlap extensively (Figure 4), leading to redundancy and potentially reduced interpretability. The adaptive nature of B-splines ensures that each MF contributes meaningfully to the overall representation.

d) Fine-Tuning Through Basis Functions: As shown in Figure 5, increasing the number of B-spline basis functions allows for more detailed modelling of the second input feature. The additional basis functions enable the MF to capture finer variations, including subtle peaks and valleys. This flexibility provides a trade-off between accuracy and interpretability, as overly complex MFs may become less intuitive. Nevertheless, the ability to adjust the number of basis functions gives practitioners greater control over the expressiveness of models.

In general, the learned B-spline MFs align with previous findings in the literature [6], demonstrating their ability to adapt naturally to real-world data. For example, the geographical and logistical constraints of installing electrical lines in rural towns lead to membership degrees that are neither strictly convex nor bounded by a full membership degree of 1. By removing traditional constraints such as convexity and normality, B-splines provide a more expressive representation of fuzzy sets, enabling deeper insights into the underlying data structure.

The observed advantages of B-spline MFs have significant implications for applications requiring data-driven modelling. Their flexibility allows for a more accurate representation of diverse phenomena, while their inherent adaptability reduces the need for manual tuning of MFs. This makes B-splines a promising alternative to traditional Gaussian or triangular MFs in tasks that involve complex or noisy data.

IV. Conclusion

This work demonstrates that employing B-splines as flexible, unconstrained MFs offers a promising data-driven alternative to traditional fixed-shape approaches. Rather than relying on conventional forms such as Gaussian, triangular, or trapezoidal functions, this approach allows the MFs to adapt naturally to the specific characteristics of the data. This flexibility enables the modelling of non-convex and sub-normal distributions that frequently arise in real-world applications, thereby enhancing the expressive power of fuzzy systems.

Our experiments across multiple datasets, including the Ele1 dataset, demonstrate that B-spline-based MFs can capture complex relationships with greater fidelity than traditional MFs. The learned B-splines were able to adjust to multiple local peaks and valleys and even exhibited a data-driven pruning behaviour that emphasised the most relevant regions of the input space. In contrast, traditional MFs, constrained by their predefined shapes, often produced overlapping regions that diminished the distinctiveness and interpretability of fuzzy sets.

These findings underscore the potential benefits of abandoning strict shape constraints in favour of a more flexible, data-driven design. By allowing MFs to take on any form suggested by the data, the proposed approach provides a more objective approximation of the underlying patterns, broadening the applicability of fuzzy models in complex scenarios.

Future research could explore dynamic or adaptive mechanisms to optimise the number of knots, their placements, or the number of MFs per input. Investigating methods for adaptive knot placement and basis function selection could further improve efficiency and accuracy. Furthermore, extending these ideas to higher-dimensional problems and more sophisticated fuzzy inference architectures, such as hierarchical models, offers exciting possibilities for tackling diverse real-world challenges.

References

[1] L. A. Zadeh, "Fuzzy sets," *Inf. Control*, vol. 8, no. 3, pp. 338–353, 1965.

[2] C.-F. Juang, M.-G. Lai, and W.-T. Zeng, "Evolutionary Fuzzy Control and Navigation for Two Wheeled Robots Cooperatively Carrying an Object in Unknown Environments," *IEEE Trans. Cybern.*, vol. 45, no. 9, pp. 1731–1743, 2015.

[3] K. Vivek, K. V. Subbarao, W. Routray, N. R. Kamini, and K. K. Dash, "Application of Fuzzy Logic in Sensory Evaluation of Food Products: a Comprehensive Study," *Food Bioprocess Technol.*, vol. 13, no. 1, pp. 1–29, 2020.

[4] J. Liu, J. Lv, H. Dinçer, S. Yüksel, and H. Karakuş, "Selection of Renewable Energy Alternatives for Green Blockchain Investments: A Hybrid IT2-based Fuzzy Modelling," *Archives of Computational Methods in Engineering*, vol. 28, no. 5, pp. 3687–3701, Aug. 2021.

[5] J. M. Garibaldi and R. I. John, "Choosing membership functions of linguistic terms," in *Proc. IEEE Int. Conf. Fuzzy Syst.*, vol. 1, 2003, pp. 578–583 vol.1.

[6] J. M. Garibaldi, S. Musikasuwan, T. Ozen, and R. I. John, "A case study to illustrate the use of non-convex membership functions for linguistic terms," in *Proc. IEEE Int. Conf. Fuzzy Syst.*, vol. 3, 2004, pp. 1403–1408 vol.3.

[7] C.-H. Wang, W.-Y. Wang, T.-T. Lee, and P.-S. Tseng, "Fuzzy B-spline membership function (BMF) and its applications in fuzzy-neural control," *IEEE Trans. Syst. Man. Cybern.*, vol. 25, no. 5, pp. 841–851, 1995.

[8] Y. Yuan and H. Zhuang, "A genetic algorithm for generating fuzzy classification rules," *Fuzzy Sets Syst.*, vol. 84, pp. 1–19, 1996.

[9] A. A. A. Esmin, A. R. Aoki, and G. Lambert-Torres, "Particle swarm optimization for fuzzy membership functions optimization," in *IEEE Int. Conf. Syst. Man Cybern.*, vol. 3, 2002, pp. 1–6.

[10] C.-F. Juang and P.-H. Chang, "Designing Fuzzy-Rule-Based Systems Using Continuous Ant-Colony Optimization," *IEEE Trans. Fuzzy Syst.*, vol. 18, no. 1, pp. 138–149, 2010.

[11] J.-S. Jang, "ANFIS: adaptive-network-based fuzzy inference system," *IEEE Trans. Syst. Man. Cybern.*, vol. 23, no. 3, pp. 665–685, 1993.

[12] C. Chen, R. John, J. Twycross, and J. M. Garibaldi, "An extended ANFIS architecture and its learning properties for type-1 and interval type-2 models," in *Proc. IEEE Int. Conf. Fuzzy Syst.*, 2016, pp. 602–609.

[13] C. Chen, C. Wagner, and J. M. Garibaldi, "Gradient-based Fuzzy System Optimisation via Automatic Differentiation – FuzzyR as a Use Case," *arXiv:2403.12308 [cs.AI]*, 2024.

[14] C. Chen, T. R. Razak, and J. M. Garibaldi, "FuzzyR: An Extended Fuzzy Logic Toolbox for the R Programming Language," in *Proc. IEEE Int. Conf. Fuzzy Syst.*, Glasgow, UK, 2020, pp. 1–8.

[15] C. de Boor, "On calculating with B-splines," *J. Approx. Theory*, vol. 6, no. 1, pp. 50–62, 1972.

[16] A. S. Weigend and N. A. Gershenfeld, *Time series prediction: Forecasting the future and understanding the past*. Santa Fe: Addison-Wesley, 1993.

[17] O. Cordón, F. Herrera, and L. Sánchez, "Solving Elec-

trical Distribution Problems Using Hybrid Evolutionary Data Analysis Techniques," *Appl. Intell.*, vol. 10, no. 1, pp. 5–24, 1999.
[18] C. Chen, R. John, J. Twycross, and J. M. Garibaldi, "Type-1 and interval type-2 ANFIS: a comparison," in *Proc. IEEE Int. Conf. Fuzzy Syst.*, 2017, pp. 1–6.