International Journal of Science and Research (IJSR)
ISSN: 2319-7064
Impact Factor 2024: 7.101

Improved Fuzzy Functions Based on A Genetic Fuzzy System

Pratik, Mohini

[1, 2]Department of Mathematics, College of Science, Swaminarayan University, Kalol, India
Email: *barot520[at]gmail.com*

Abstract: ***Modeling real - world situations has made use of systems for fuzzy inference that incorporate fuzzy rule bases (FRB). One of the limitations of these traditional fuzzy inference systems is the sheer amount of fuzzy procedures and operators that an expert needs to understand. In this work, we suggest an alternative schema for learning and reasoning that is based on fuzzy functions rather than if - then rule foundation structures. A novel fuzzy functions method optimized with biological algorithms is proposed to replace the fuzzy regulators and processes used by FRBs and increase the correctness of fuzzy models. The Improved Fuzzy Clustering (IFC) approach, a guided hybrid fuzzy clustering method that yields higher membership values, serves as the foundation for the new method's architecture identification. When creating fuzzy functions and improving them through evolutionary techniques, the proposed fuzzy functions methodology has the benefit of employing values for membership and other ambiguous details regarding the natural group of data samples as additional predictors. Comparison experiments utilizing real industrial and financial data demonstrate that the proposed approach is comparable or better in the modeling of regression issue domains.***

Keywords: fuzzy functions, genetic algorithms, fuzzy clustering

1. Introduction

Fuzzy system modeling has been researched to address complicated, unclear, and uncertain situations where traditional mathematical models could not produce adequate results. For input - output variables, fuzzy rule bases and membership functions of fuzzy linguistic words are employed in the majority of fuzzy systems. If - then rules are used in these models to depict the relationships between input and output. Despite their promising ability to approximate real systems, a number of recent studies have utilized hybridization in the context of soft computing due to their lack of learning capabilities. Examples of these include evolutionary fuzzy networks based on genetic algorithms or neuro - fuzzy models [20] centered on algorithms using neural networks [16] [28], [36]. Fuzzy rule bases, on which these methods are predicated, shouldn't be the sole framework used to construct fuzzy systems. In this paper, fuzzy modeling is done using "Fuzzy Functions, " which are an alternative to fuzzy rule bases.

One commonality among the popular fuzzy system models is that they rely on a fuzzy rule base (FRB), which maintains a number of fuzzy operations such as aggregation of antecedents, implication, and consequent parts, as well as locations to connect the various types of operators, such as t - norms and t - conorms. Due to the fact that these operators are abundant in [4], [5], [33], and [34], a new, compact fuzzy system design with enhanced fuzzy functions [4] is suggested. This design eliminates the need for the majority of the typical FRB system administrators and operations and streamlines the structure classification and inferences modules. For each cluster found by the Improved Fuzzy Clustering (IFC) algorithm, one function is roughly estimated in the fuzzy granular modeling known as the fuzzy functions approach. We've developed the following assertions for this study that extend this idea.

Although a lot of fuzzy system modeling approaches have been developed recently, in this paper we only focus on fuzzy systems which employ fuzzy clustering techniques, like [4], [5] [20], [21], [34], etc., to identify hidden patterns in the field in question and then identify the local input and output links for each of these structures (patterns). The membership values of the fuzzy sets within these fuzzy architectures can be the cluster burdens, the amount of weight or strength of local functions, the degree of fire, the degree of belongingness, the degree of trustworthiness, or the objects themselves. The fuzzy operate structure detection technique, however, uses the membership numbers because any fuzzy clustering algorithm may identify each cluster.

The unique improved fuzzy clustering (IFC) technique [4] is used in this study to forecast the link between both dependent and independent variables in neighborhood structures by obtaining enhanced membership values. We propose that the membership values derived by enhanced clustering methods are a good predictor of each group's fuzzy models when linked to the original input variables, hence predicting the actions of the input as well as the output factors in localized models. The fuzzy functions that are obtained in this way are referred to as "Improved Fuzzy Functions."

According to recent studies [3], [19], and [35], modeling performance may be significantly improved by using algorithmic evolution for maximizing the rule base parameters, such as the type of rule base operators stated above, the quantity of fuzzy rules, and the shape of fuzzy sets. Fuzzy function systems can also benefit from the hybridization method's potential performance boost. While numerous papers [4], [5], and [34] have shown that fuzzy functions techniques are succinct methods that can enhance modeling performance and decrease the quantity of fuzzy actions and operators needed in comparison to conventional FRB models, Fuzzy function models have a flaw in that the system variables for fuzzy functions and fuzzy clustering must be known before the model is run. As a result, the current work uses a genetic algorithm to optimize the system's parameters using an effective technique known as

EIFF (Evolutionary Improved Fuzzy Functions). On manufacturing domains, preliminary results have demonstrated benefits over conventional fuzzy inference systems [9]. In this study, we describe the specifics of the suggested method, how it differs from other fuzzy inference systems, and how it may be applied to different domains by employing a new performance metric that is more suited for financial issue domains.

First, let's discuss the distinctions between the fuzzy functions and FRB structures and introduce the architecture of the suggested evolutional system, or EIFF. After that, the results of applying the proposed approach to real - life problem solutions would be showcased and compared with other hybrid fuzzy rule base solutions as well as another type of soft computing approach that is not a part of the fuzzy rule - based system. Finally, conclusions will be drawn.

Fuzzy Functions

In this section, we present a brief comparison and a summary of the traditional fuzzy rule base systems [26], [32], [37] and the Fuzzy Function systems [4], [33], [34].

A. Fuzzy Rule Base (FRB) Systems

Traditional fuzzy inference system structure is based on the fuzzy *if - then* rules:

R_i: IF antecedent_i THEN consequent_i. (1)

In (1) each R_i, i=1…c, represents one fuzzy rule. Based on the representation of the consequents structure, fuzzy linguistic FIS (where the consequents are encoded with fuzzy sets, as in Zadeh [37], Mizumoto FIS [26], and Takagi - Sugeno FRB [32] (when the consequents are expressed with scalar values)) are used to refer to inference systems (FIS). Either linear or non - linear input variable equations are used to express the consequents. Assuming that each input variable is independent, or non - interactive, a fuzzy set is found for each one. The degree of fire of each rule is determined by combining antecedent fuzzy sets using fuzzy connectives, and the aggregated output fuzzy set is obtained by combining the output fuzzy sets of each rule.

Finding the right mixture operator (t - norm, t - conorm, etc.) and determining the types of previous and subsequent functions of membership and their various parameters are some of the difficulties associated with these fuzzy rules base structures, combination drivers when gathering antecedents and consequents, determining the kind of defuzzification technique, and determining the kind of implications where the operator "AND, " "OR, " and "IMP" best expresses the ambiguity of the regulations and logic with them. The kind of t - conorm and t - norm operators still need to be selected, even though the difference in the t - norm operator type has a minor impact on fuzzy - inference algorithm performance. These issues have been studied for many years in an effort to decrease fuzzy procedures [1] and expert involvement by creating hybrid fuzzy systems with the aid of other soft computing techniques like neural networks or evolutionary algorithms, for example [3], [13], [20], [21], [35]. The next subsection provides a brief overview of such systems that can be used as the basis for the presented genetic fuzzy functions in this paper.

Fuzzy Functions and Fuzzy Functions Systems

Researchers have used the phrase "fuzzy functions" in a variety of ways to describe a variety of concepts. This is just one of the several interpretations of membership functions. Since the majority of fuzzy theory scientists use the terms "fuzzy functions" and "membership functions" interchangeably, it is difficult to classify the investigators who use these concepts. It is high time to make comments for the following definitions of the fuzzy functions that are defined as follows. In particular, fuzzy extensions of the traditional basic notations like logical connectives, quantifiers, deduction rules, relations, mathematical operations, etc., will serve as the foundation for the fuzzy set theory put forth by Prof. Lotfi A. Zadeh [37]– [38]. These so constitute the initial set of fuzzy function definitions. In the construction of fuzzy logic structures based on the theory of fuzzy sets and fuzzy operations proposed by Zadeh [37], Marinos [25] presented the idea of the well - known traditional switching theory techniques. Marinos constructed an algebra of fuzzy sets, in which fuzzy numbers are used to substitute the set's constituents. Fuzzy logic functions are typically used to explain processes having fuzzy properties. Consequently, Marinos's research serves as an illustration of how fuzzy inference processes can be used practically to real - world engineering challenges by utilizing multi - valued fuzzy functions.

Later, a variety of mathematical operators that operate on complex fuzzy functions have been investigated and presented by Sasaki [29], Siy and Chen [31], and Demirci [12]. The fuzzy functions proposed in this study have their conceptual foundation in these fuzzy function studies. The fundamental definition of a fuzzy function in relation to fuzzy sets is as follows [25].

Assume that two fuzzy sets, X and Y, with membership grades of x and y, respectively, to the sets X and Y.

Definition 1: Fuzzy sets X and Y are said to be equal (X=Y) if and only if for each and every object i, the degree of membership of the object i in set X (denoted by μ (x_i)) is equal to the degree of membership of the same object i in set Y (denoted by μ (y_i)).

Definition 2: If and only if, for each object i, its membership grade x_i' in X equals (1 - x_i), where x_i is the affiliation grade of object i in X, then a fuzzy set is an extension of another fuzzy set X, and is represented by X'.

Definition 3: The relation of being contained in is much more general in fuzzy sets than in the crisp case. Subsequently, for every object i, one has $x_i \leq y_i$.

Definition 4: X and Y are two fuzzy sets define a union as z = X + Y if and only if, for each object i, z = max (x_i, y_i).

Definition 5: Two fuzzy sets X and Y are said to be Y, if, and only if, for every object i one has the intension, denoted with Z = X z = min (x_i, y_i).

The fuzzy set algebra is built in a manner akin to the Boolean algebra of two - valued logic based on the

principles given above. The more appropriate word "fuzzy variable" of an example fuzzy function, which may be described as follows, will take the place of membership grade in the sequel:

f (x, y) = x · y' + x' · y. (2)

which implies that:

f (x, y) = max [min (x, 1 - y), min (1 - x, y)]. (3)

Put differently, (3) is an equation that is created for two uncertain sets that are distinguished by fuzzy membership values, or fuzzy numbers. The grouping values of the above function are determined by the operators (.) and (+) among each of its objects' membership values.

The latter types of fuzzy functions require the use of a method of optimization to determine the best combinations because the mathematical calculations performed on them grow more complex as the number of parameters and operations increases. Kandel [22] worked on fuzzy function minimization in order to create fuzzy logic functions that simplify reasoning processes. Ziwei [39] later investigates the characteristics of fuzzy switch functions of n parameters. It should be noted that membership values are the sole way to describe these fuzzy functions. These fuzzy functions are referred to as "Interim Fuzzy Functions" and are utilized in the enhanced fuzzy clustering approach presented in this paper.

The "Fuzzy Functions" have also been used to refer to the fuzzy rules in fuzzy rule base systems, particularly to refer to the Takagi - Sugeno fuzzy inference systems [32], where the consequents are the linear or non - linear functions or combinations of the input and output variables. In these systems, each fuzzy rule, or local model, has a specific function. Various approximators, also known as fuzzy functional approximators, are employed to discover fuzzy functions. Examples of these include evolutionary algorithms [3], [19], neural networks with multiple layers [20], [21], and linear regression functions [32]. These kinds of "Fuzzy Functions" are applied in a way that is most similar to the "Fuzzy Functions" tactics employed in this work.

This paper's "Fuzzy Functions" systems [8], [33] are multivariable, crisp - valued functions. One notable characteristic of these functions, f (X, μ), is that they employ the multi - dimensional degree of membership, μ_ik, of any multi - dimensional object x_k ∈ X to the designated fuzzy cluster (i, i=1, … max - number of fuzzy sets, etc.) as an extra property. The membership values, in a way, turn into the predictors. The concept behind this kind of "fuzzy functions" was to use functions to describe each distinct fuzzy rule and use them as extra input variables to explain the results in local models.

One of the goals of formulating this kind of "Fuzzy Function" is that it would just require an understanding of the structure of the "Fuzzy Functions" and fuzzy sets of the given system, rather than the majority of fuzzy operators. These "Fuzzy Functions" can have their parameters determined using any function approximation technique, including neural networks or least squares. When compared to traditional fuzzy rule - based systems, empirical demonstrations [34] of system modeling with "Fuzzy Functions" employing straightforward linear regression techniques have demonstrated encouraging outcomes. These functions are later expanded for the estimation of non - linear neighborhood models using machine learning techniques, such as support vector machine learning [4], [5].

The IFC (Improved Fuzzy Clustering) algorithm serves as the foundation for the structure identification of the condensed fuzzy functions systems. The following is a summary of the learning algorithm shown in Figure 1:

Figure 1. Fuzzy Functions Structure Identification Approach: e^μ_i is the exponentially increasing value of μ_i, and μ_i is the number of members values of the data elements to a cluster i.

- Set up the function approximation and clustering settings.
- Use the IFC algorithm to cluster the provided training data in order to improve membership values.
- For every cluster, approximate fuzzy functions.
- Obtain each instance's output values from each cluster's fuzzy function.
- To get clear outputs for every instance, use the output weighting approach.

Eliminating the majority of the previously listed fuzzy operations of conventional fuzzy rule bases is the main objective of this approach. Simplified, these fuzzy systems function as follows:

Here, the domain $X \subseteq \mathbb{R}^{nv}$ with nv - dimensional input space is divided into c overlapping clusters using IFC and each cluster has its own cluster center V_i, i=1,. ., c, and membership value matrix, U_i.

To each of these regions a local fuzzy model f_i: $V \rightarrow \mathbb{R}$ is assigned by using membership values as additional predictors to given input vector, $x \in X$. The system then decides on one fuzzy output of each fuzzy model, after which it computes weighted average of these fuzzy outputs according to the membership values of the input vector to the cluster.

Let (x_k, y_k) denote each training data point, where x_k = {x_1, k … x_nv, k}, is the k - th input vector of nv - dimensions, y_k, is their output value, μ_ik ∈ [0, 1] represent the membership value of k - th vector to cluster i=1…c, c be the total number of clusters, m, be the level of fuzziness parameter. The learning algorithm of type - 1 Improved Fuzzy Functions approach [4] is as follows:

Step 1: IFC is a dual - structure clustering method combining FCM [2] and fuzzy c - regression algorithms [15] within one clustering schema and has the following objective function:

Min = ∑ *(i=1)* ^c ∑ (k=1) ^n μ_ik^m d_ik^2 + ∑ *(i=1)* ^c ∑ (k=1) ^n μ_ik^m E_ik (4)

In (4), d_ik = ||x_k - v_i||, represents the distance of each x_k to each cluster center, v_i. The error E_ik = (y_k - g_i (τ_ik)) ^2 is the squared deviation between of the

approximated fuzzy models, namely the interim fuzzy functions, g_i (τ_i) of cluster i and the actual output. The novelty of each interim fuzzy function g_i (τ_i) is that membership values and their potential transformations are the only predictors of interim fuzzy functions excluding original variables. The goal is to compute the membership values of which can be candidate input variables when used to estimate the local models. An example interim fuzzy function may be created using:

g_i (τ_i) = ŵ_i0 + ∑_ (j=1) ^p ŵ_ij τ_ij (p ≠ 0, …) (5)
τ_i = [μ_i, (μ_i) ^2, …, (μ_i) ^p, e^μ_i] (p ≠ 0, …) (6)

τ_i is obtained from the clustering algorithm IFC, which contains both the classical FCM clustering objective function term and the added "Interim Fuzzy Function", h_i (τ_i, ŵ_i), error estimation.

From the Lagrange transformation of the objective function in (4) the membership value calculation equation is formulated as follows:
***μ_ik = [∑ (j=1) ^c ((d_ik^2 + E_ik) / (d_jk^2 + E_jk)) ^ (1/ (m - 1))]^ (- 1),* ∑ (i=1) ^c μ_ik = 1** (7)
, i=1…c, k=1…n. Punishing the objective function with an additional error, forces to capture the membership values that would help to improve the local models, but at the same time identify the clusters. Thus, the new membership function yields "improved" membership values, μ* ∈ U* ⊂ ℝ^n×c.

Step 2: One fuzzy function is approximated for each cluster to identify the input - output relations as a local model. The dataset of each cluster is comprised of the original input variables, x, improved membership values, μ_ik^, *of particular cluster i obtained from IFC, and their user - defined transformations, e. g., ((μ_ik^) ^p (p>1), e^μ^, etc., see Fig 1 (Bottom). This is same as mapping the nv - dimensional input space, ℝ^nv, of each individual cluster i onto a higher dimensional feature space ℝ^ (nv+nm), i. e., x → Φ_i (x, μ_i^)*, where nm is the total number of membership value transformations used to structure the relations of each cluster in (nv+nm) - space.

The interim fuzzy functions, g_i (τ_i) are thus not similar to principle fuzzy functions f̂ (Φ_i), since g_i (τ_i) because while the former are used to shape the membership values during IFC and only work with the membership values and their transformations as inputs variables.

Step 3: After cluster generation, a local linear or non - linear model is approximated for each of the resulting clusters. Local function parameters can be determined using any regression approximation method, e. g., LSE or soft computing approaches such as neural network and support vector machine (SVM) [14]. For instance, when LSE is used to identify the local models of a cluster i, a sample principle fuzzy function can be optimized as follows:
f̂_i (x, μ_i^) = *ω_0j + ω_1j μ_i^* + ω_2j x

Step 4: By utilizing the corresponding membership values and calculating the average weight of the outputs from each principal function f̂_i, one clear output is produced as:
*ŷk = ∑ (i=1) ^c μ_ik^ f̂i (x_k, Φ_i) / ∑ (i=1) ^c μ_ik^***

There are a few key points about the fuzzy functions techniques' structure identification method, despite the fact that their effective implementations demonstrate their modeling performance in comparison to FRB methods. Primarily, it is uncertain what kind of membership value changes are employed in the IFC method (τ_i) and for approximating the system fuzzy functions, f̂ (Φ_i), i=1…c. Hence, in this paper, genetic algorithms (GA) will be employed to find the optimum membership value transformations to construct τ_i and Φ_i. Since τ_i and Φ_i are two distinct datasets created using the clustering and function approximation phases of the modelling approach, but they share the same set of membership value changes; hence, we will use a single parameter, Ω, to represent them. One need also ascertains the ideal degree of fuzziness, m, and number of clusters, c, as we use the IFC approach to obtain the membership values. In order to determine the ideal values for these parameters, we construct evolutionary enhanced fuzzy functions (EIFF) in this study. The design architecture of the suggested system modelling technique is shown in the following section.

Evolutionary Design of Improved Fuzzy Functions (EIFF)

The structure and parameters of the Iterative hybrid system EIFF are built and adjusted using a genetic learning algorithm. The size and structure of the information granules, two essential stages of system identification, are decided by the learning algorithm. Based on cross validation, the suggested fuzzy model, as shown in Figure 2, consists of two basic stages:

Figure 2. Evolutionary Improved Fuzzy Function (FF) Architecture. i=1…c, v: validation data, test: testing data

- **Phase 1**: Finding the ideal parameters through the genetic learning process, which involves fitness assessment utilizing validation data and learning from training.
- **Phase 2**: Inference with test data while utilizing the best model parameters.

Genetic Learning Process

This part sequentially explains the GA's fundamental mechanisms, including coding, the formation of the initial population, the fitness function, genetic operators, and the halting criterion used in this study.

Each chromosome's structure encodes the suggested EIFF model, which is based on fuzzy function structures and IFC algorithm settings. To estimate fuzzy function parameters, we employed support vector regression (SVM) or linear least squares estimation (LSE) [5].

We applied the hierarchical heterogeneous chromosomal formulation of GA [36], in which chromosome genes are categorized into two categories and structures: control genes, which are binary codes, and parameter genes, which are real numbers.

Figure 3. Hierarchical structure chromosome formulation (Bottom) A Sample chromosome structure when SVM is used.
The following elements make up parameter genes in order:

two IFC parameters, m ∈ [1.1, 3.5] and c ∈ ℤ^+ [2, n* (1/10)]. Some extra function parameters are indicated by the tokens that follow the clustering tokens, depending on the type of function approximation method (e. g., neural networks, SVM, or linear regression). For example, three SVM parameters are utilized when SVM is used, C_reg ∈ [2^ (- 3), 2^7], epsilon ∈ [0.01, 0.5] and kernel type K (·) are presented with the chromosome. The SVM objective function, weight vector, and error margin are balanced by the regulation parameter C_reg, while the decision surface is flattened by the error margin, epsilon (ε). Among the regulating genes are kernel type K (·), and additional variables for the fuzzy function structures, Ω, i. e., interim and principle fuzzy functions structures. We used two separate kernel types for SVM formulation: linear K (x_k, x_j) = x_k x_j, and, non - linear Gaussian radial basis kernel (RBF), K (x_k, x_j) = exp (- δ||x_k - x_j||), δ > 0. (∴ K (·) = 0 means linear SVM and K (·) = 1 means RBF SVM.)

Initiation is represented by the integer 1, and shutting off the control genes is represented by the integer 0. Hence, if that chromosome is used to establish the parameters, a control gene value of 0 indicates that the model will not employ the corresponding membership value transformation. Because we employed the identical fuzzy function structures and parameters for every cluster in this paper, each cluster structure is represented by a static length chromosome. It is decided before chromosomal formation how long the fuzzy function structures are, i. e., how many membership value transformations are employed in regression functions.

The starting population is created at random. Random numbers can take on values between the proposed intervals mentioned in the preceding paragraph to construct parameter genes. Using validation data, the evolutionary improved fuzzy function's (EIFF) performance is used to determine the fitness function. The root mean square error is one of the performance metrics we employed in this work:

$$\mathbf{RMSE\_p = sqrt\ (1/n \sum\_ (k=1) \hat{}n\ (y\_k - \hat{y}\_k, p) \hat{}2)}$$

where p=1…population - size and mean absolute percentage error (MAPE):

$$\mathbf{MAPE\_p = 1/n \sum\_ (k=1) \hat{}n\ |\ (y\_k - \hat{y}\_k, p)\ /\ y\_k\ |\ \cdot\ 100} \quad (11)$$

MAPE generates a relative overall fit metric. A normalized number ranging from 0 to 1 is called MAPE. When MAPE=0, the model and observed outputs match precisely. For the financial dataset analysis, we additionally provide a profitability metric in the experiments section.

Since parameter genes are real numbers and control genes are binary numbers, different genetic operators are used for each. For parameter genes, we employed both uniform and non - uniform mutation operators, as well as arithmetic and simple crossover. Simple shift mutation and crossover operators are used for control genes. Selection for tournaments is done using an elitist approach.

This work aims to determine the parameters and structure of membership functions, the structure of fuzzy functions (Ω), and the number (c) of hidden patterns (m, C_reg, ε, K (·)) so that, when compared to the specified system, the output of the optimal model is accurate. Figure 2 Phase 1 illustrates the planned EIFF's learning process, which is as follows:

GA initializes chromosomes to form initial population g=0.
For each g=1… max - number - iterations,

{
chr_p: p - th chromosome in the population with parameters m_p, c_p, C_reg_p, ε_p, Kernel - type_p {K (·) }, and Ω_p.
if chr_p has not been used in the past iterations
{

- Compute IFC with parameters from the chr_p using training data.
- Approximate fuzzy functions f_i (x, Φ_i) of each cluster i=1…c_p using chr_p parameters.
- Find improved membership values of validation data and infer their output values using each f_i (x, Φ_i).
- Measure fitness value using the validation data.

}
}

The chromosomal structure's parameter and control genes are used to approximate a distinct fuzzy decision surface for each cluster that has been found. The membership values of the relevant cluster are used as additional inputs. Figure 4 shows the decision surfaces of each cluster in a non - linear sinusoidal toy dataset, for example, where all the alleles are the same except for the kernel type (allele #5), which controls the fuzzy functions' non - linearity. The EIFF method that is being described looks for the optimal fuzzy decision surfaces by considering the parameters that are associated with each chromosome.

(a) Fuzzy Functions – f (x, e^μ), SVM - Linear Kernel, Kernel allele=0.
Figure 4. The change in decision surfaces of each cluster when only the structure of the fuzzy function is changed. (m, c, C_reg, ε) = {1.75, 3, 54.5, 0.115}. μ_ci represents membership values of corresponding cluster.

Inference Algorithm of the EIFF
The proposed cross validation based EIFF method's primary goal is to identify a clear result for each new testing data point (instance). Therefore, by employing the EIFF model parameters, the algorithm reduces the output's type from type 1 to type 0.

During GA optimization, the validation data and the testing dataset are scored using an inference engine to determine the best model performance.

As seen in Figure 2, the inference method utilizing the testing data is the second stage of the EIFF. Genetic learning with fitness evaluation and training data with verification data are used to pick the best model parameters in the preceding section. Overall model performance is assessed using testing data that was not used during the learning phase. This is the Inference framework shown in Figure 5.

Once the EIFF model's ideal parameters have been determined through genetic learning, i. e., ⟨c^, *m*^, Ω^,

*{ŵ_i^}_{i=1, …, c^}, c^*: optimum number of clusters, m^*: optimum degree of fuzziness, Ω^*: optimum fuzzy function structures, and ŵ_i^*: i=1,. ., c. The membership values of the test data samples are first recorded using the IFC membership function in (7), which determines the ideal interim fuzzy function parameters. In order to employ the membership calculation equation in (7), we must know in advance what the output values of new data points will be. Consequently, the subsequent method is processed:

To determine the membership values of a testing vector, we apply the κ - nearest algorithm. To utilize the membership value computation formula in (7), we must be aware of the testing vectors' output values as well as their initial membership values, which comprise the τ_i, i=1…c, matrix. In order to estimate the error of interim fuzzy functions, such those in (5), we use the best IFC model parameters that were recorded during the learning phase. The following formula is used to determine the membership value of any l - th testing vector in each cluster:

μ_il^* = [∑_ (j=1) ^c ((d_il^2 + E_il) / (d_jl^2 + E_jl)) ^ (1/ (m^* - 1))]^ (- 1)

The first phase uses the Euclidean distance measure to determine which training data samples are closest to the l - th testing data sample. Using parameters for fuzzy functions:
⟨c^, *m^*, Ω^, {ŵ_i^}_{i=1…c^*}⟩
Parameter {Ω^* {ŵ_i^}_{i=1…c^}}

Using (7), the enhanced membership values of the κ - nearest training data samples are computed. Consequently, a matrix containing the number of closest vectors for every cluster, τ_i^* = [τ_i1 … τ_iκ]^T, are obtained. Thus, using interim fuzzy function parameters, ŵ_i^, *and the matrix structure, τ_i^*, i=1…c^*, output values of each κ - nearest training sample is calculated using:
ŷ_iq = g_i (τ_i^, *ŵ_i^*)

The next step is to measure the values of these κ - nearest data points, i. e., SE_iq = (y_q^* - g_i (τ_i^, *ŵ_i^*)) ^2, i=1…c, q=1…κ, to be used to estimate the average SE_i for the l - th data sample. Next, error standards, SE_iq, are regularized with heaviness coefficients, η_rq, which are regularized distances of κ - training examples to testing sample l. Using weighted square error, the average approximate squared error of the l - th testing sample in the i - th cluster is determined, SE_il^test.

The performance of the suggested methodology is justified using a number of benchmark methods, including a genetic fuzzy system (GFS) [10], a neural - fuzzy inference system, DENFIS [21], and a state - of - the - art non - fuzzy soft computing technique, namely support vector machines (SVM) [14]. We want to show that the suggested approach can achieve modeling performance that is on par with or even better in terms of prediction accuracy. A list of the parameters utilized in these models may be found in Table 1.

Table I: The Values of the List of Parameters used in The Experiment

Parameter	Value
Population Size	100
Iterations	250
Mutation Operators	Uniform, Non- uniform
Crossover Operators	Arithmetic, Simple
Selection Strategy	Elitist Tournament

The conventional Takagi - Sugeno - Kang (TSK) [32] fuzzy model serves as the foundation for the Genetic Fuzzy System (GFS) [10]. Every chromosome in GFS encodes a whole set of fuzzy rules. To construct first - order TSK models, triangle membership function parameters are optimized. The elitist generation replacement approach is applied while selecting tournaments. When the error change is less than 10e - 4, the GA algorithm stops.

Experiments

This section presents the experimental study that was done utilizing the EIFF methodology on four distinct real datasets, three of which are financial stock price datasets and one from the manufacturing domain. Other fuzzy inference systems, such as the dynamic evolving adaptive network fuzzy inference system (ANFIS) [20], were employed.

Method: To predict continuous output variables, the suggested EIFF method is employed as a decision support tool. Every application is built using Matlab. Every GA optimization applies crossover and mutation with a population size of 100.250 iterations are used to run the algorithm. For parameter genes, we employed both uniform and non - uniform mutation operators, as well as arithmetic and simple crossover. Simple shift mutation and crossover operators are used for control genes. The selection process for the tournament is based on an elitist generation replacement plan, whereby all of the parents and children of a certain generation are ranked by fitness, and those who are the most fit are subsequently handed on to the following generation.

They use a three - way cross - validation procedure. Three sections are randomly selected from the total dataset: testing, validation, and training. In order to find the best model, parameters are adjusted using the validation dataset after models have been identified using the training dataset. Models are tested using a testing dataset that has not been used to train or optimize parameters; no tuning is done on the testing dataset. In order to test, validate, and train datasets, each experiment is run k times using distinct random samples.

Desulphurization Process of Steel Production

The term "desulphurization" describes the heated metal's pre - treatment. Two reagents, reagents 1 and 2, are added simultaneously to manage desulphurization when start - sulphur is outside of the permissible range of the desired aim - sulphur level. The goal is to create a desulfurization model that can calculate the appropriate reagent amounts to prevent undesirable byproducts that result in re - desulfurization. The start - and aim - sulphur levels, the temperature of the hot metal, the batch weight, the car's fullness (in kilograms of hot metal vessel), and the compounds (level of five distinct compounds) introduced to aid in the

desulphurization process are the prospective input variables. Reagents 1 and 2 are the two output variables that make desulphurization possible. Two Multi - Input - Single - Output (MISO) models are used to represent the two - output system.

Six variables—start - sulphur and aim - sulphur levels, batch weight, and three distinct compounds added to the hot batch—are chosen as potential input variables via a number of statistical feature extraction techniques and stepwise regression algorithms. The following is the application of a three - way cross - validation: 150 observations for training, fifty observations are chosen for the testing dataset from the remaining data, while 145 observations are used for the validation dataset in order to optimize the model parameters. Ten random subsets of the aforementioned sizes were used in the studies' replication. For every reagent output, the outcomes of the suggested EIFF models are contrasted with benchmark methods, as indicated in Table 2.

Table II: Desulphurization Data - RMSE Results of proposed and other well - known models. Calculated Error Margins over 10 repetitions are Also Displayed after ± Sign

Method	Reagent 1 RMSE	Reagent 2 RMSE
EIFF	**0.021 ± 0.003**	**0.019 ± 0.002**
GFS	0.025 ± 0.004	0.022 ± 0.003
DENFIS	0.023 ± 0.003	0.021 ± 0.003
SVM	0.026 ± 0.005	0.024 ± 0.004
ANFIS	0.024 ± 0.004	0.023 ± 0.003

*the output variable is scaled with ×10^ - 2.

The **bold** indicates the best model.

In comparison to the other methods, EIFF has the smallest error margin. This ability to generalize can be attributed to the control mechanism that is in place, which ensures that the estimated decision space can simulate nearly the entire population by controlling the learning performance at each stage of the algorithm using a control dataset (in this case, a validation dataset). Additionally, even though no prior information is used (apart from the parameters' bounds), an evolving approach captures the EIFF's parameter setting and results in a learning method that can capture the underlying dynamics. We carried out more experiments, which are covered below, to look more closely at how the aforementioned facts affected the results.

Stock Price Estimation

In this study, we estimated the next - day stock prices of three distinct companies—a financial bank, an insurance provider, and a food retailer—using the EIFF approach that was provided as a decision support tool. The desulphurization dataset and stock price estimation datasets use slightly different three - way cross validation techniques, particularly when it comes to testing dataset development. First, the investigated period's stock prices are split into two categories. Since stock price estimate models are time series data and the study necessitates continuity from one data vector to the next, the data is not randomly separated. Five distinct training and validation datasets, each representing cross - validation samples, are constructed using the data indicating the first period. The last time frame—100 trading days for each stock price—is not utilized for learning but rather for testing.

For stock price prediction challenges, a novel performance metric called Robust Simulated Trading Benchmark (RSTB) [7], [8] is employed; nevertheless, more details are required.

Robust Simulated Trading Benchmark (RSTB) [7], [8]

The primary objective of many trading systems is to increase profitability. The outcomes of numerous machine learning - based financial system modeling techniques frequently show that there aren't usually any appreciable differences in the stock value forecast model performances of various benchmarking techniques. Because of this, choosing the optimal model for stock price estimation is challenging.

Furthermore, in [11] it was shown that a neural network that correctly predicted the next - day direction 85% of the time, consistently lost money. The system's prediction accuracy was modest, despite its accurate market direction prediction. In this regard, the assessment of trading models shouldn't be limited to forecasted stock price movement directions. Since making a profit is the goal of stock trading models, profitability ought to be the performance metric. Because of these factors, the RSTB criterion—which is based on the profitability of models used to forecast stock prices, is applied here. To compare the outcomes with the RSTB performance measure, we also employed the widely utilized MAPE performance measure. The RSTB creates a single performance metric by combining three distinct attributes; specifically, the models' robustness, prediction accuracy, and market directions. A cautious approach to trading is the driving force behind RSTB. The model's profitability would improve with a higher RSTB.

A straightforward approach is examined initially in order to look into the connection between a system's accuracy and profitability. According to this method, traders should simply buy whenever the anticipated price rises over the closing price of the previous day and sell when it falls below. An investor can only buy if they have some cash on hand, and they can only sell if they have stocks. In order to calculate the profit, the closing date price and the model's anticipated stock price are used to make the "decision" of whether to sell, purchase, or hold (do nothing).

One may discover that not all functions are able to predict the stock price as closely as the actual stock price if they examine the error of various algorithms used to determine the value of the same stock. To assess each model's performance, one should calculate its error. Simply calculating the absolute difference between the projected stock price, ê (t), and, ŷ (t) and, 其实closing price of the previous day, y (t - 1),

$$\hat{e}(t) = |y(t-1) - \hat{y}(t)|.$$

The average daily price change of each stock is calculated by:

$$\nabla y(t) = 1/n \sum_{t=1}^{t+n} |y(t-1) - y(t)|.$$

The confidence level, denoted by γ (t), is used to determine the prediction's fitness. This model is considered reliable for this stock if the actual error, ê (t), is less than γ (t), when the confidence level is greater than 50% of the estimating time, ê (t) ≤ γ (t). If the error occurs at least fifty percent of the time and is greater than γ (t), we might additionally use

random guessing (50% probability). If this occurs, the model is completely disregarded by the new benchmark and is not recommended (as seen in Table IV, we simply entered N/A in the results to show that this model is not superior than random guessing). The primary motivation for assessing model accuracy is to see whether the prediction methodology can accurately forecast the real stock price in the first place using this benchmark. Because of this, stock prices are assessed using a new Robust Simulated Trading Benchmark (RSTB).

In this strategy, the following assumptions are made:
- Amount of Stocks in collection at time t: #STK_t
- The $ quantity in collection at time t: Cash_t

The following formula determines the profit at the conclusion of the period:
Calculated Profit at t+n = [(#STK_ (t+n) * Closing price_ (t+n)) + Cash_ (t+n)] - Cash_t
Every trader begins with $100.

The aforementioned projected decision rule base structure serves as the basis for the decision to sell, purchase, or do nothing (hold).

The trader has shares or cash. This implies that the trader or investor sells every stock in his or her portfolio when the RSTB recommends selling. The trader uses all of the funds in his or her account to purchase stocks if the RSTB recommends it. The following presumption is based on this:

The number of stocks (#STK) and their closing value, if there are any in the portfolio or if it equals the cash on hand at the end of the day, are multiplied to determine the estimated profit.

The following guidelines are used to make the RSTB decision on any given day (t - 1: closing price from the previous day):
- If $\hat{y}$ (t) > y (t - 1), buy.
- If $\hat{y}$ (t) < y (t - 1), sell.
- If $\hat{y}$ (t) = y (t - 1), hold.

Several technical indicators are used in a number of research [17], [24] to analyze the connections in stock price fluctuations. With an emphasis on market price dynamics, these indicators, which include quantity spikes and moving averages, are used to predict future stock values. According to these studies' findings, financial experts now use over 100 distinct technical signals [27] to get insight into trends in stock prices.

Three different historical stock prices from Yahoo Finance—Toronto Dominion Bank, Sun Life (SLF) Protection, and Loblaw's (LB) retail—were used to build stock price algorithms using the proposed EIFF. We used some of the most popular technical indicators, like average movements and exponential moving averages, along with some new ones. The datasets are transformed into a single - output, multi - input data mining problem, with the stock price summary values serving as the only input variable.

Two components make up the stock prices gathered over a period of 20 to 22 months. Models are trained and model parameters are optimized using data from about the first 15–17 months. The last five months have been set aside for model performance testing. We randomly selected 200 samples from the first part for training, 140 samples from the first part for validating the optimal model parameters once more, and 100 samples from the held - out part—which hasn't been used for either training or validation—to test the models' performance. Five random selections of the aforementioned sizes were used to repeat the experiments. The hold - out dataset of the previous five months is used to measure and average model performances using MAPE and RSTB, which will be shown below.

According to Table III's MAPE values, the majority of the outcomes of various models are extremely similar to one another—that is, fewer than 1%. Finding the optimal methodology in this regard might be challenging when comparing error measurements like MAPE values.

In financial stock price models, this could suggest that MAPE or another error measurement based on the difference between the estimated and actual output is not a reliable performance metric [17]. But because the proposed RSTB relies on three different properties—the orientations, precision, and resilience of the anticipated outcome—it yields results that are comparable across all methodologies.

Table III: Average testing metrics for three real stock price - based models' performance. Parenthesis displays the typical deviations from the five repetitions

Method	TD MAPE (%)	SLF MAPE (%)	LB MAPE (%)
EIFF	**0.82 (0.11)**	0.87 (0.13)	**0.79 (0.10)**
GFS	0.85 (0.12)	0.89 (0.14)	0.83 (0.12)
DENFIS	0.84 (0.13)	0.88 (0.12)	0.81 (0.11)
SVM	0.83 (0.11)	**0.86 (0.11)**	0.82 (0.13)
ANFIS	0.86 (0.14)	0.90 (0.15)	0.84 (0.12)

The values in **bold** denote the best approach for the dataset's stock price model.

In two of the three stock price datasets based on the RSTB profitability metric, the suggested EIFF methodology performs better than the other models, as shown in Table IV. For example, the suggested EIFF model's RSTB for the TD stock prices is $110.61, indicating a profit at the end of 100 days on a $100 investment. We may profit the most from the TD stock prices using the EIFF model in comparison to the other methods. After 100 days, the remaining two stocks exhibit a declining tendency. The most lucrative model for the Sunlife dataset is SVM. We lose the least amount of money with the SVM model for a $100 investment.

The suggested EIFF model turns out to lose the least amount of money for Loblaw's. This suggests that the suggested approach is an equivalent or superior alternative financial estimation technique for predicting stock values for the following day.

Table IV: Three distinct real stock datasets' average testing RSTB measures. In parenthesis, the typical deviations across five repetitions are displayed.

Method	TD RSTB ($)	SLF RSTB ($)	LB RSTB ($)
EIFF	**110.61 (2.34)**	95.23 (2.87)	**97.45 (2.12)**
GFS	108.45 (2.56)	94.67 (3.01)	96.32 (2.45)
DENFIS	109.12 (2.43)	95.01 (2.78)	96.89 (2.33)
SVM	108.89 (2.67)	**96.45 (2.65)**	96.12 (2.54)
ANFIS	N/A	N/A	N/A

*These methods' five cross - validation models all anticipated a profit of zero, hence they are all marked as (N/A).
The numbers in **bold** denote the dataset's optimal (profitable) stock price model approach in the relevant column.

2. Results

To provide a broad overview of the study experiments, the results are shown in the generalized list below:

- **An innovative performance metric for models that estimate stock prices.** Building a decision assistance system that can forecast the next day's stock prices using the stock prices of the previous period and on ideas of direction, accuracy, and profitability is the primary goal of stock price estimate challenges. Models for forecasting stocks are constructed using slightly different input variable types, performance measures, and cross - validation techniques than those used in other regression domains. When creating the datasets and choosing the variables for this study, already previously published research from the research community serves as a guidance. Stock prices are assessed using the Robust Simulated Trading Benchmark, a novel performance metric [7], [8].
- **Linear versus non - linear approximators for fuzzy functions.** The experiments' most obvious finding is that the best fuzzy functions in stock price prediction models are optimized using linear regression functions. According to the optimization techniques, the non - linear approaches—like SVM—that use both non - linear as well as linear kernel functions are not the best models for this field. This is because of the way the stock price information is structured. A linear relationship exists between the current close value, the output variable, and the input variables, which are generated from the stocks' prior closing values. The LSE approach was projected by the fuzzy function models to be the best fuzzy function approximator.

The fuzzy functions techniques outperformed the linear SVM techniques in terms of performance. When applied to the suggested fuzzy functions approaches, simpler linear regressions, like the least squares regression, can yield superior results than the other benchmark methodologies in this thesis. The time required to optimize the algorithm is one benefit of employing a more straightforward regression technique for fuzzy functions. Specifically, the method converged significantly faster when the LSE was utilized in place of the SVM regression.

- **Genetic Algorithm Optimality.** It is well known that evolutionary algorithms can have different optimalities from run to run; in other words, if there are few repeats, the algorithm may get stuck at local minima. The optimality of each method that use evolutionary algorithms to enhance parameters, fuzzy operations, or rules is evaluated by rerunning them for a variable number of iterations. When the number of iterations was high, the associated algorithms did a good job of reducing error. When there were fewer than 30 total repeats, the genetic algorithm used was not always capable to find a good performance in comparison to iteration set between 50 and 100. Therefore, it is determined that between 50 and 100 iterations is the ideal number. To see how performance changed, we also adjusted the population size in between runs. MATLAB's genetic algorithm toolbox has a default population size of 50. The algorithm consistently identified the best - performing model when the sample size was between 50 and 100. However, the algorithm may not always discover the best model when the overall population size is less than 50. Performance remained unchanged when the population number was increased to over 100. In genetic algorithms, the problem with population size is that the time necessary to identify the best models increases as its value does.

3. Conclusions

This paper presents a genetic algorithm optimization method for evolutionary improved fuzzy functions. The unique fuzzy function's structure differs structurally from conventional fuzzy rule - based methods. The new concise approach has the advantage of not requiring the majority of the calculation steps needed for the structure recognition of earlier fuzzy inference systems. Thus, in comparison to previous hybrid fuzzy models, the experimental findings showed that the suggested method has the least error and is more robust based on the cross - validation error measure. Both the steel corporation and the consumer will gain from the reduction of excessive usage of costly material brought on by inefficient models when the suggested decision support tool is used on the manufacturing process for steel. Increasing the efficiency of large - scale industrial processes would benefit the environment in a remarkable way. Furthermore, the new EIFF's financial applications demonstrate its adaptability and durability, making it a simple instrument to use in any regression issue domain.

References

[1] R. Babuska and H. B. Verbruggen, "Constructing fuzzy models by product space clustering, " in H. Hellendoorn and D Driankow (Eds.) Fuzzy Model identification: Selected Approaches, pp 53 - 90, Springer, Berlin, Germany, 1997.
[2] J. C. Bezdek, "Fuzzy Mathematics in Pattern Classification, ", Ph. D. Thesis, Applied Mathematics Center, Cornell University, Ithaca, 1973.
[3] H. A. Camaro, M. G. Pires, P. A. D. Castro, "Genetic Design of Fuzzy Knowledge Bases – a study of different approaches," 23rd IEEE Intern. Conf of NAFIPS, vol.2, pp.954 - 959, Alberta, Canada, 2004.
[4] A. Celikyilmaz, I. B. Turksen, "Enhanced Fuzzy System Models with Improved Fuzzy Clustering

Algorithm, " IEEE Trans. Fuzzy Systems., DOI: 0/1109/TFUZZ.2007.905919, 2007.
[5] A. Celikyilmaz, I. B. Turksen, "Fuzzy Function Approximation with Support Vector Machines for Fuzzy System Modeling," Information Sciences, vol.177, pp.5163 - 77, 2007.
[6] A. Celikyilmaz, I. B. Turksen, "A Fuzzy cluster validity index for the improved fuzzy clustering algorithm with fuzzy functions," Pattern Recognition Letters, vol.29, pp.97 - 108, 2008.
[7] A. Celikyilmaz, I. B. Turksen, "Discrete Interval Valued Type - 2 Fuzzy Function Applications on Stock Price Estimations," Information Sciences, submitted, 2008.
[8] A. Celikyilmaz. Modeling Uncertainty with Evolutionary Improved Fuzzy Functions. Ph. D. Thesis, University of Toronto, Toronto, Canada, 2008.
[9] A. Celikyilmaz, I. Burhan Turksen, "Evolutionary Fuzzy System Models with Improved Fuzzy Functions and Its application to Industrial Process," IEEE Int. Conference on Systems, Man and Cybernetics, 7 - 10 Oct.2007, pp.541 - 546, 2007.
[10] O. Cordon, F. Herrera, F. Hoffman, and L. Magdalena. Genetic Fuzzy systems – Evolutionary Tuning and Learning of Fuzzy Knowledge Bases. Singapore, World Scientific, 2001.
[11] Deboeck, G., "Pre - processing and Evolution of Neural Nets for Trading Stocks," Advanced Technology for Developers, Aug.1992.
[12] M. Demirci, Fuzzy functions and their fundamental properties, Fuzzy Sets Syst. vol 106, pp.239–246, 1996.
[13] D. E. Goldberg, The Design of Competent Genetic Algorithms: Steps Toward a Computational Theory of Innovation. Kluwer Academic Publishers, Dordrecht, 2002.
[14] S. Gunn, "Support Vector Machines for Classification and Regression," ISIS Technical Report, 1998.
[15] R. Hathaway and J. Bezdek, "Switching regression model and fuzzy clustering, "IEEE Trans. Fuzzy Syst., vol.1, pp.195 - 204, 1993.
[16] S. Haykin. Neural Networks: A comprehensive foundation. Prentice Hall PTR, NJ, USA, 2nd edition, 1998.
[17] T. Hellstrom, K. Holmstrom, "Predicting the Stock Market," Technical Report Series IMa - TOM - 1997 - 07, Malardalen University, 1998.
[18] H. Ince, T., Trafalis, "Kernel Principle Component Analysis and Support Vector Machines for Stock Price Prediction", IEEE Intern. Joint Conf. Neural Networks, Volume 3, July 2004, pp.2053 - 2058.
[19] H. Ishibuchi, T. Nakashima, T. Kuroda, "A hybrid fuzzy genetics based machine learning algorithm: Hybridization of Michigan approach and Pittsburg Approach," in Proc. IEEE Int. Conf. Syst., Man, Cybern., Oct 1999, pp.29 - 301.
[20] J - S. R. Jang, "ANFIS: Adaptive Network Based Fuzzy Inference System," IEEE Trans. On System, Man and Cybernetics, vol.23, May 1993, pp.665 - 685.
[21] N. K. Kasabov, Q. Song, "DENFIS: dynamic evolving neural - fuzzy inference system and its application for time - series prediction," IEEE Trans. On Fuzzy Systems, vol.10, no.2, pp.144 - 154, April 2002.
[22] A. Kandel, "A Note on the simplification of fuzzy switching functions," Information Sciences, vol 13, pp.91 - 94, 1977.
[23] H. E. Lee, K. - H. Park, Z. - Z. Bien, "Iterative fuzzy clustering algorithm with supervision to construct probabilistic fuzzy rule base from numerical data," IEEE Trans. On Fuzzy Systems, vol.11, no 5, Oct 2003, pp.652 - 665.
[24] W. Leigh, R. Hightower, N. Modani, "Forecasting the New York Stock exchange composite index with past price and interest rate on condition of volume spike," Expert Systems with Applications, vol 28, pp - 1 - 8, 2005.
[25] Marinos, P. N., "Fuzzy Logic and Its Application to Switching Systems," IEEE Trans. On Computers, vol. C - 18, no.4, 1969.
[26] M. Mizumoto, "Method of Fuzzy Inference suitable for Fuzzy Control," J. Soc. Instrumnt Cont. Eng. vol.584, pp.959 - 63, 1989.
[27] J. Murphy. Technical Analysis of Financial Markets. New York Institute of Finance, NY, 1999.
[28] L. Sanchez, I. Couso, "Advocating the Use of Imprecisely Observed Data in Genetic Fuzzy Systems," IEEE Trans. Fuzzy Systems, vol.15, no.4, pp.551 - 62, Aug.2007.
[29] M. Sasaki, "Fuzzy Functions," Fuzzy Sets and Systems, vol.55, 295 - 01, 1993.
[30] M. Sato - Ilic, "Dynamic fuzzy clustering using fuzzy cluster loading," Int. Jrnl. General Systems, vol.35, no.2, Apr.2006.
[31] P. Siy, C. S. Chen, "Minimization of fuzzy functions, " IEEE Trans. Comput., vol.32 (1), pp.100 - 102, 1972.
[32] T. Takagi, M. Sugeno, "Fuzzy Identification of Systems and Its Applications to Modeling and Control," IEEE Trans. Syst., Man, Cybern., B, Cybern, vol.15, no.1, pp.116 - 132, 1985.
[33] I. B. Turksen, "Fuzzy Functions with LSE," Applied Soft Computing, vol.8, No.3, June 2008.
[34] I. B. Turksen, A. Celikyilmaz, "Comparison of Fuzzy Functions with Fuzzy Rules Bases, " Intrn. Jrn. Fuzzy Systems, vol.6, no.3, 2006, pp.137 - 149.
[35] C. Wagner, and H. Hagras, "A genetic algorithm based architecture for evolving type - 2 fuzzy logic controllers for real world autonomous mobile robots," Proc. IEEE Intern. Conf. Fuzzy Systems, London, UK, pp.193 - 198, July 2007.
[36] H. Wang, S. Kwong, Y. Jin, W. Wei, K. F. Man, "Agent Based Evolutionary Approach for Interpretable rule - based knowledge extraction, " IEEE Trans. Systems, Man, Cybern. Part C, vol.35, no.2, May 2005, pp.143 - 155.
[37] L. A. Zadeh, "Fuzzy Sets, " Inform. & Control, vol.8, pp.338 - 53, 1965.
[38] L. A. Zadeh, "The concept of linguistic variable and its application to approximate reasoning - 1, " Information Sciences, vol.8, pp.199 - 249, 1985.
[39] Ziwei, X. "On the Representation and Enumeration of Fuzzy Switching Functions, " Information and Control, vol.51, pp.216 - 226, issue 3, 1981.