# Optimization of interval type-2 fuzzy system using the PSO technique for predictive problems

## Dinh Sinh Mai , Trong Hop Dang & Long Thanh Ngo

Published online: 30 Oct 2020.

Submit your article to this journal ⬀

View related articles ⬀

View Crossmark data ⬀

Taylor & Francis
Taylor & Francis Group

🔓 OPEN ACCESS | Check for updates

# Optimization of interval type-2 fuzzy system using the PSO technique for predictive problems*

Dinh Sinh Mai [a], Trong Hop Dang[b] and Long Thanh Ngo [a]

aLe Quy Don Technical University, Hanoi, Vietnam; bHanoi University of Industry, Hanoi, Vietnam

**ABSTRACT**

An interval type-2 fuzzy logic system (IT2FLS) can function well with uncertain data, with which a type-1 fuzzy logic system (T1FLS) is ineffective because its membership function rests upon crisp values. However, similar to T1FLSs, there are challenges associated with IT2FLSs in selecting parameters, which can significantly affect the accuracy of the classification results with their relatively high sensitivity. This paper discusses and proposes a hybrid model based on IT2FLS and particle swarm optimization (PSO) for prediction problems. The main objective of this paper is to find the optimal solution for the unknown fuzzy systems using labelled data for training the fuzzy system. The PSO technique was used to find the optimal parameters of the Gaussian membership functions which utilized for IT2FLSs. The authors tested two data sets for each of the two prediction problems, namely: burnt forest area prediction and wine quality prediction. The predictive results were compared with other predictive methods including random forest (RF), support vector machines (SVM), artificial neural network (ANN), adaptive neuro fuzzy inference system (ANFIS) and IT2FLS with parameters generated by using the fuzzy c-means algorithm (IT2FLS-FCM). Experiment results showed that the proposed method could significantly improve accuracy compared to several other predictive techniques.

## 1. Introduction

Prediction is a field of study on the likelihood that certain events will occur based on scientific analysis of the data collected. Predictions are used in a wide variety of areas, each with its own specific prediction requirements, which calls for an array of different prediction methods. Despite the vast challenges in analysing and predicting a problem, the significance of this task cannot be overstated. In order to make meaningful predictions, it is necessary to rely on data collection and processing to determine the future trend of phenomena and scenarios based on several mathematical models. The accuracy of prediction models depends significantly on the data set used for training which usually contains errors and noise.

---

With its advantages in processing noise data and uncertain data, fuzzy logic has been applied in many areas of life. One of the widely used applications of fuzzy logic is the fuzzy system. Fuzzy systems can effectively solve problems in automatic control, prediction, classification, etc.

Fuzzy systems are based on type-1 fuzzy sets because they have crisp membership function values which make it difficult to describe the uncertainty of the data fully. Therefore, over the past few years, there have been a number of studies on the subject, including: type-1 fuzzy set (T1FS) to interval-valued fuzzy set (IVFS) (Dzung et al. 2015; Maciel & Ballini 2019), type-2 fuzzy set (T2FS) (Emanuel & Patricia 2019) and interval type-2 fuzzy set (IT2FS) (Das et al. 2015; Long et al. 2012).

T2FSs are characterized by a three-dimensional fuzzy MF, including the footprint of uncertainty (FOU) that can directly model and handle the uncertainty of data (Mendel 2007). Once the type-1 MF is selected, all uncertainties will be eliminated because the type-1 MFs are entirely correct (Mendel & Liu 2013). The type-2 fuzzy logic system (T2FLS) based on the T2FS has been used in many practical applications such as predictive problem (Abbas et al. 2012), industrial control (Das et al. 2016; Nguyen & Saeid 2016), data classification (Liang & Mendel 2000). One of the cases of T2FS is more widely used, which is an interval type-2 fuzzy system (IT2FS) (Mendel et al. 2006).

The paper proposes an optimal model based on the interval type-2 fuzzy logic system and PSO technique to predictive problems. This work is expanded from the ACIIDS 2020 paper (Mai et al. 2020) with the aim of providing a general IT2FLS-PSO model that can be used for prediction purposes. The proposed model can be utilized to select the optimal parameters for unknown IT2FLSs.

The paper is organized as follows: Section 1 introduces the research problem; Section 2 reviews some related research; Section 3 presents some background knowledge; Section 4 describes an optimal model for the IT2FLS based on the PSO technique, and Section 5 reports results of the experiments and Section 6 includes conclusion and direction for future research.

## 2. Related work

Uncertainty exists in almost any collected data (Mai & Long 2015). There are many causes for data uncertainty, such as errors during collection or effects of environmental conditions (Long et al. 2015). Fuzzy logic has the advantage of handling uncertainty data which has been widely applied in many problems including clustering and classification (Maciel & Ballini 2019; Mai et al. 2018b; Nguyen & Saeid 2016; Patricia & Oscar 2014; Trinh & Mai 2019).

There have been many studies on improving fuzzy clustering algorithms based on spatial information, possibilistic information (Mai et al. 2018a), semi-supervised methods (Mai & Long 2018). However, many issues arise when fuzzy clustering algorithms of type-1, type-2 and interval type-2 are applied in problems where the data is highly nonlinear (Mai 2018; Mai & Long 2019). The use of fuzzy systems is more robust when dealing with uncertain data and highly nonlinear data modelling capabilities.

The fuzzy classification system has the advantage of being able to estimate any nonlinear function with the appropriate number of fuzzy rules (Mai et al. 2020). This has been

proven through many practical applications with good predictability by the fuzzy system. However, the fuzzy systems based on the type-1 fuzzy set (T1FS) are not suitable for handling noise or uncertainty in data when crisp membership functions (MFs) are involved (Mendel 2017).

Recently, fuzzy systems based on type-2 fuzzy sets have been widely used in practical problems with better ability to deal with uncertain data than fuzzy sets based on type-1 fuzzy sets. Due to the high computational complexity of type-2 fuzzy sets, fuzzy systems based on interval type-2 fuzzy sets are more preferable (Emanuel et al. 2020; Thanh & Saeid 2015).

Some studies and applications of type-2 fuzzy logic in pattern recognition, classification and clustering problems can be found in Patricia and Oscar (2014). A design methodology for IT2FLS with Centre-Of-Sets defuzzification is proposed in Juan et al. (2020) with the use of complementary information to determine the uncertainty limits of membership functions. Emanuel et al. (2019) introduce a shaded type-2 fuzzy inference system, which can provide a good approximation for general type-2 fuzzy inference systems while reducing computational costs. Das et al. (2015) proposed an evolving interval type-2 neural fuzzy inference system to achieve rapid and accurate inference, a data-driven interval-reduction approach to convert interval type-1 fuzzy set in antecedent to type-1 fuzzy number in the consequent. This method evolves automatically and adapts its network parameters using a metacognitive learning mechanism concurrently. However, most methods do not allow automatic adjustment of the parameters, and this may result in a failure to achieve the desired results.

The problem of fuzzy predictive control of nonlinear networked control systems subject to parameter uncertainties and defective communication links is studied by Lu et al. (2015). An interval type-2 Takagi–Sugeno (TS) fuzzy model is employed to describe the nonlinear plant subject to parameter uncertainties, which can be captured with the lower and upper membership functions. Panella and Rizzi (2003) propose a fuzzy system to control vehicle traffic flows on a street network. The core of the system is constituted by ANFIS, which is used to predict the time series represented by the fuzzy membership of traffic measures to the three predefined flow states. Alreshoodi et al. (2016) propose a prediction model for the perceptual quality of wireless 4kUHD H.265 video streaming based on IT2FLS.

In the studies (Ayad et al. 2019; Chao et al. 2016), extended ANFIS architecture was proposed by incorporating an additional layer to the blurring process, extended architecture can fit both type-1 and type-2 fuzzy models. Lin et al. (2014) described a self-developed type-2 fuzzy neural network using type-2 fuzzy sets in the premise and Takagi–Sugeno–Kang (TSK) type as a consequence of the fuzzy rule. Li et al. (2018) improved the Mamdani and TSK fuzzy inference systems using the interval type-2 TSK fuzzy rule bases. The operation of the IT2FLS is more complicated than that of its type-1. Some suggestions for designing the IT2FLSs are introduced in Wu and Mendel (2019) to make IT2FLSs more accessible to IT2FLS designers. Although there are various approaches in handling uncertain data, it is apparent that most studies on IT2FLSs are faced with challenges determining the optimal parameters (Amit et al. 2020).

In the case of designing the IT2FLSs for particular applications, the use of bio-inspired optimization methods can assist in finding the appropriate parameter values and structure of the fuzzy systems. Optimization techniques including genetic algorithms, PSO

and ant colony optimization in IT2FLS have been widely used in many fields (Oscar & Patricia 2012). Another observation is that the use of the PSO algorithm (Kennedy & Eberhart 1995) to optimize the FOU of fuzzy systems provides an exciting alternative to parameter generation in T2FLS (Shihabudheen et al. 2018). Eduardo et al. (2019) use computational intelligence methods to create hybrid models as a classification method. This hybridization can increase performance in fuzzy systems to solve complex problems. Several biological-inspired methods are also used to optimize type-2 fuzzy inference system parameters. Among them, genetic algorithms and particle swarm optimization are commonly used (Fernando et al. 2016).

Despite the capability of previous methods in handling uncertainty in processing real-world data, major obstacles exist with regard to the selection of optimal parameters. The optimization of the antecedent parameters of the IT2FLS, the Cuckoo Search (CS) and Genetic Algorithms (GAs) are applied in Gonzalez et al. (2016) for image edge detection. The dynamic parameter adaptation methodology for Ant Colony Optimization (ACO) based on IT2FLSs is presented in Frumen et al. (2017). Besides, to increase the accuracy of IT2FLSs, Oscar and Patricia (2019) use type-2 intuitionistic fuzzy models for pattern recognition applications. When working with fuzzy systems, there are many ways to construct fuzzy membership functions, and the Gaussian function adopted by this study is one of them. In fact, each type of function has advantages and disadvantages, and the selection is based on experience.

Review of current literature on approaches to handling uncertain data indicates that when working with fuzzy systems, rather than being static, the parameters vary greatly depending on the specific problem or even specific data set. Therefore, using optimization techniques to find the optimal parameters for each issue will improve the stability and efficiency of fuzzy systems. This paper uses the PSO technique to find the optimal parameters for the membership function of unknown IT2FLS.

The next section will introduce some background knowledge about interval type-2 fuzzy logic system and PSO technique.

## 3. Background

### 3.1. Interval type-2 fuzzy logic system

The IT2FLS is characterized by interval type-2 fuzzy set (IT2FS) (Mendel 2017; Mendel et al. 2006) consisting of five main parts as shown in Figure 1.

There are two types of IT2FSs as shown in Figure 1(a,b), one of which is type-reduction and then defuzzification, the second is direct defuzzification. However, in practice the type combining both type-reduction and defuzzification is more widely used because of lower computational complexity.

The IT2FLS works as follows: the crisp inputs are the attributes of the initial data, which fuzzifier into the input IT2FSs and then activate the inference engine and rule base to maps input IT2F sets into output IT2FSs. These output IT2FSs are then processed by the type-reducer to obtain T1FSs (type reducers). The defuzzifier then defuzzifies output T1FSs to create the crisp output.

Details of IT2FLS components are described as follows:
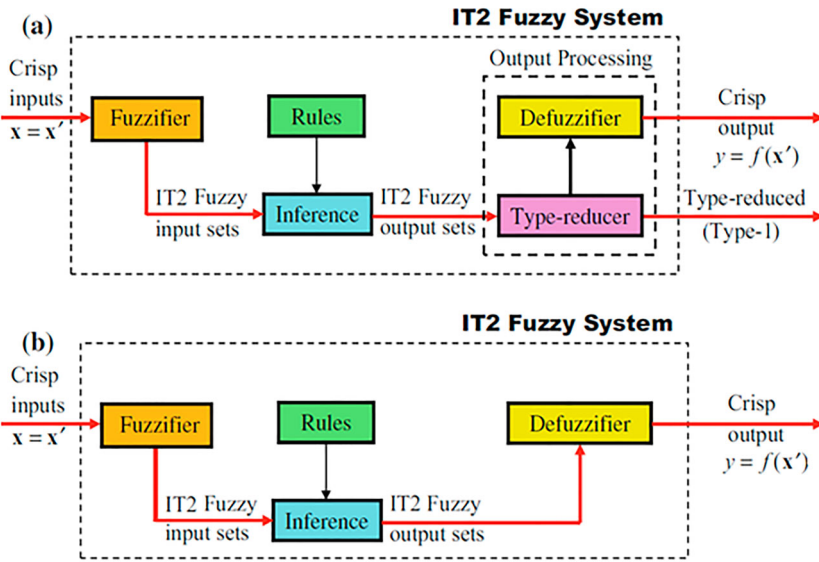
- + Fuzzifier:

**Figure 1.** Interval type-2 fuzzy logic system architecture (Mendel 2017).

With T1FS, two types of fuzzifiers are used as singleton and non-singleton, meanwhile with T2FS, there are three types of fuzzifiers used including singleton, type-1 non-singleton and IT2 non-singleton. The fuzzifier maps a crisp input which will depend on the choice of the type of fuzzifier. Assuming there are $n$ inputs $X = (x_1, x_2, \ldots, x_n) \in X_1 x X_2 x \ldots x X_n$, $\tilde{A}_x$ is a set of type-2 fuzzy inputs. For example, if $\tilde{A}_x$ is a type-2 fuzzy singleton fuzzifier, then $\mu_{\tilde{A}(x_i)} = 1/1$ when $x_i = x_i'$ and $\mu_{\tilde{A}(x_i)} = 1/0$ when $x_i \neq x_i'$ and $x_i \in X_i$.

- + Rule Base

Consider the input $x_1 \in X_1, x_2 \in X_2, \ldots, x_n \in X_n$ and $c$ output $y_1 \in Y_1, y_2 \in Y_2, \ldots, y_c \in Y_c$. The rules of T2FS are similar to those of T1FS, with the exceptions of the antecedents and consequents: T1FS is replaced with T2FS:

$$R^i: \text{ IF } x_1 \text{ is } \tilde{F}_1^i \text{ and } \ldots \text{ and } x_n \text{ is } \tilde{F}_n^i$$
$$\text{THEN } y_1 \text{ is } \tilde{G}_1^i \text{ and } \ldots \text{ and } y_c \text{ is } \tilde{G}_c^i \tag{1}$$

with $M$ is the number of rules in the rule base, $i = 1, \ldots, M$.

- + Fuzzy Inference Engine

The inference engine and the rules that allow the mapping from input T2FS to the output T2FS. Each rule in a fuzzy rule base with $M$ rules having $n$ inputs $x_1 \in X_1, x_2 \in X_2, \ldots, x_n \in X_n$ and output $y_k \in Y_k$, they can be written as follows:

$$R_k^i : \tilde{F}_1^i x \tilde{F}_2^i x \ldots x \tilde{F}_n^i \rightarrow \tilde{G}_k^i = \tilde{A}^i \rightarrow \tilde{G}_k^i \tag{2}$$

where $\tilde{F}_j^i$ is the $j$th T2FS, $j = 1, \ldots, n$, which is defined by a lower and upper bound membership function:

$$\mu_{\tilde{F}_j^i}(x_j) = [\underline{\mu}_{\tilde{F}_j^i}(x_j), \bar{\mu}_{\tilde{F}_j^i}(x_j)], \quad i = 1, \ldots, M; \; k = 1, \ldots, c \quad (3)$$

Compute the firing interval of the $i$th rule, where $*$ denotes the product operation:

$$\begin{aligned} \underline{f}^i(x) &= \underline{\mu}_{\tilde{F}_1^i}(x_1)*\underline{\mu}_{\tilde{F}_2^i}(x_2)*\cdots*\underline{\mu}_{\tilde{F}_n^i}(x_n) \\ \bar{f}^i(x) &= \bar{\mu}_{\tilde{F}_1^i}(x_1)*\bar{\mu}_{\tilde{F}_2^i}(x_2)*\cdots*\bar{\mu}_{\tilde{F}_n^i}(x_n) \end{aligned} \quad (4)$$

- + Type Reduction

   There are several algorithms used in type reduction, such as Karnik–Mendel algorithm (KM), Enhanced Karnik–Mendel algorithm (EKM), iterative algorithm and stopping condition (IASC), enhanced IASC algorithm (EIASC) (Mendel 2017).

- + Defuzzification

   The final crisp value of output of the IT2FS model is calculated by combining the corresponding outputs of $M$ rules. For defuzzification solution, we calculate the average left most point and right most point, therefore the crisp output for each output is calculated as follows:

$$Y_k(x) = \frac{y_{kl} + y_{kr}}{2}, \quad k = 1, \ldots, c \quad (5)$$

## 3.2. Learning method

The PSO is an adaptive evolution algorithm based on finding the optimal solution for the population. The idea of algorithms comes from the hunting behaviour of the birds (Kennedy & Eberhart 1995). Each problem will converge at one or several optimal solutions in the search space, considering each particle is a particle and a set of particles will be a population.

Each state of the population in the search space is considered a candidate solution. The optimal solution is found by moving particles in the search space according to the position and velocity of the particle as the following equation:

$$\begin{aligned} vt_i^{k+1} &= \omega*vt_i^k + c_1*r_1*(P_{i\text{best}} - v_i^k) + c_2*r_2*(G_{i\text{best}} - v_i^k) \\ v_i^{k+1} &= v_i^k + vt_i^{k+1} \end{aligned} \quad (6)$$

in which $v_i^k$ is the position of particle $i$th in $k$th generation, $vt_i^k$ is the velocity of particle $i$th in $k$th generation, $\omega$ is the coefficient of inertia, $c_1$, $c_2$ is the acceleration coefficient, with a value of 1.5–2.5; $r_1$, $r_2$ is the random number, with values in the range of [0, 1].

In each loop, the optimal position search is performed by updating the velocity and position of the particle. For this study, the target value of each particle location is

determined by a fitness function as follows:

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \bar{y}_i)^2} \qquad (7)$$

The use of the PSO algorithm in fuzzy systems is as follows:

**Algorithm 1** The PSO algorithm

**Step 1**: Initialize population and parameters $\omega$, $c_1$, $c_2$, $r_1$, $r_2$, $T_{max}$.
**Step 2**:
2.1 **For** each particle:

- Calculate fitness function value by Equation (7).
- **If** the fitness function value is better than the best fitness function value $P_{best}$ in history:
   **Set** the current fitness function value as the new best value $G_{best}$.

2.2 Choose the particle with the best fitness value of all the particles as the $G_{best}$.
2.3 **For** each particle:
   Calculate particle velocity and update particle position according the equation (6).

2.4 **If** (number of loops $> T_{max}$) **or** (minimum error criteria is attained) **then** Stop, **else goto** Step 2.1.
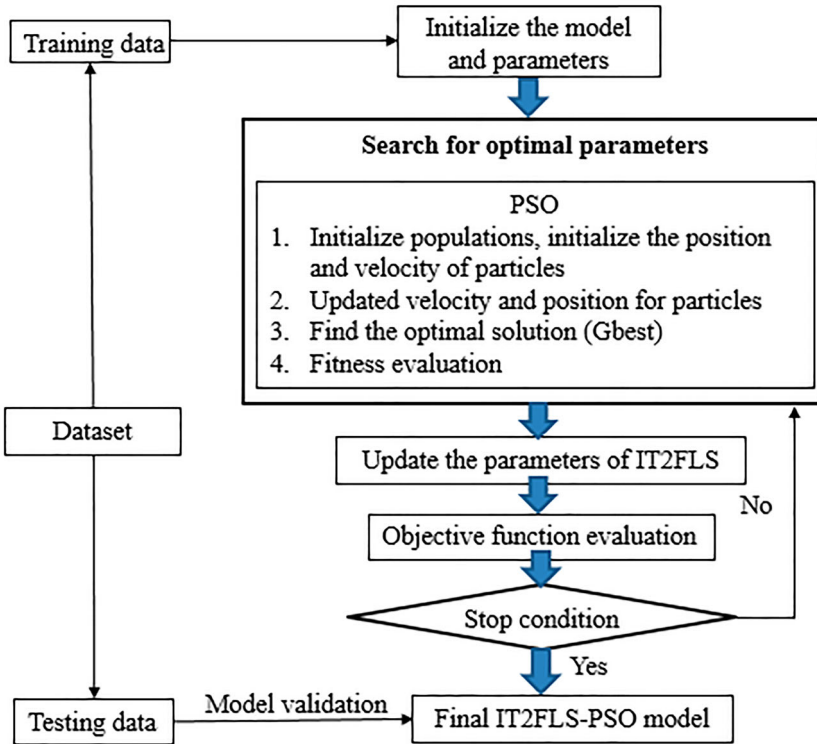


**Figure 2.** Framework of IT2FLS-PSO model.

## 4. The optimal model of the interval type-2 fuzzy system is based on PSO technique

### 4.1. IT2FLS-PSO model

In this section, an optimal model between the interval type-2 fuzzy logic system and PSO technique is proposed. The diagram of the IT2FLS-PSO model is shown in Figure 2.

Dataset is sampled and divided into a training data set and validation data set. The initial parameters will be initialized based on experience. Still, to increase the efficiency of the model, these parameters can be initialized using some simple learning techniques such as clustering algorithms.

Inputs are crisp values that will be fuzzified using membership functions. The general architecture of IT2FLS model also includes five layers, in which the type-1 fuzzy membership functions in T1FLS are replaced with the interval type-2 fuzzy membership functions.

In this paper, the IT2 Gaussian membership function is used to create IT2 fuzzy membership values. The IT2 Gaussian membership function for the interval type-2 fuzzy inference system is described as follows:

$$\mu_{\tilde{F}_j^i}(x_j) = \exp\left(-\frac{1}{2}\left(\frac{x_j - m_j^i}{\sigma_j^i}\right)^2\right) = N(x_j, m_j^i, \sigma_j^i) \tag{8}$$

with $\sigma_j^i \in [\sigma_{j,1}^i, \sigma_{j,2}^i]$ controlling the shape of the IT2 Gaussian membership function (see Figure 3). Fuzzy membership values for inputs are created using the membership function.

In the stage of finding optimal parameters, the PSO technique is used, wherein each PSO loop will give the potential parameter set for IT2FLS-PSO model.

The EIASC algorithm is used to defuzzification because they are easy to setup and the computational complexity is smaller than the other algorithms. The next step is to compute the output interval of the $k$th fuzzy rule for the output, which is an interval
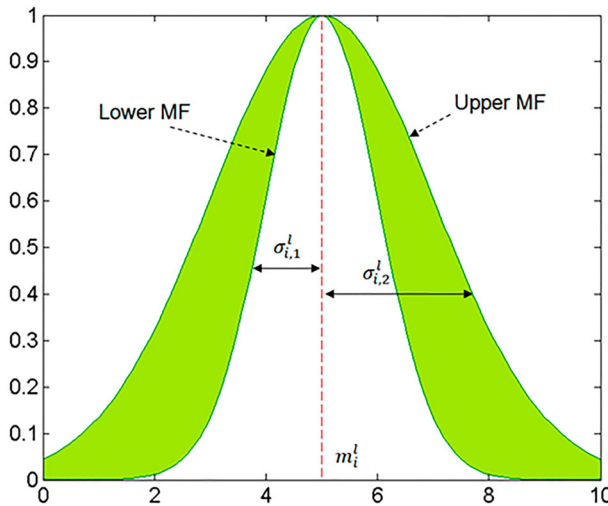


**Figure 3.** The Gaussian membership function for the interval type-2 fuzzy logic system (Mendel 2017).

T1FS $y_k = [y_{kl}, y_{kl}]$, the steps for calculating left most output $y_{kl}$ and right most output $y_{kr}$ using the EIASC algorithm are detailed below.

EIASC algorithm for calculating $y_{kr}$ is as follows:

**Algorithm 2** EIASC algorithm to find the $y_{kr}$ centroid

**Input**: Dataset $X = \{x_k, x_k \in \mathsf{R}^M, \ k = 1, \ldots, n\}$, the number of clusters $c(1 < c < n)$, fuzzifier parameters $m_1, m_2, m$.
**Output**: The centroid matrices $y_{kr}$.
**Step 1**: Without loss generality assume that sort $n$ patterns on each of $M$ features in an ascending order:
$x_1 \leq x_2 \leq \cdots \leq x_n$ ($\bar{u}_{ik}, \underline{u}_{ik}$ will also change the order corresponding to $x_1 \leq x_2 \leq \cdots \leq x_n$).
**Step 2**: Initialize $a_i = \sum_{k=1}^n \bar{u}_{ik} x_k$; $b_i = \sum_{k=1}^n \bar{u}_{ik}$; $t = n$.
**Step 3**: Compute
$a_i = a_i + x_t(\bar{u}_{it} - \underline{u}_{it})$; $b_i = b_i + (\bar{u}_{it} - \underline{u}_{it})$
$y_{kr} = a_i/b_i$; $t = t - 1$
**Step 4**: If $y_{kr} > x_t$ stop, else go to Step 3.

The steps taken to find the $y_{kl}$ centroid are similar to those in Algorithm 2, with modifications in steps 3, 4 and 5. The EIASC algorithm to find the $y_{kl}$ centroid is described in detail as follows:

**Algorithm 3** EIASC algorithm to find the $y_{kl}$ centroid

**Input**: Dataset $X = \{x_k, x_k \in \mathsf{R}^M, k = 1, \ldots, n\}$, the number of clusters $c(1 < c < n)$, fuzzifier parameters $m_1, m_2, m$.
**Output**: The centroid matrices $y_{kl}$.
**Step 1**: Without loss generality assume that sort $n$ patterns on each of $M$ features in an ascending order:
$x_1 \leq x_2 \leq \cdots \leq x_n$ ($\bar{u}_{ik}, \underline{u}_{ik}$ will also change the order corresponding to $x_1 \leq x_2 \leq \cdots \leq x_n$).
**Step 2**: Initialize $a_i = \sum_{k=1}^n \underline{\mu}_{ik} x_k$; $b_i = \sum_{k=1}^n \underline{u}_{ik}$; $t = 0$.
**Step 3**: Compute
$t = t + 1$; $a_i = a_i + x_t(\bar{u}_{it} - \underline{u}_{it})$; $b_i = b_i + (\bar{u}_{it} - \underline{u}_{it})$
$y_{kl} = a_i/b_i$
**Step 4**: If $y_{kl} \leq x_{t+1}$ stop, else go to Step 3.

The RSME objective function will evaluate the IT2FLS-PSO model with each of these parameter sets. If the conditions are satisfied, it will be the final IT2FLS-PSO model for prediction. Otherwise, PSO will be performed again to find the optimal parameters.

The final crisp value of the output of the IT2FLS model is calculated by combining the corresponding outputs of $M$ rules. This crisp output value will be the final predicted value.

## 4.2. Training the IT2FLS model using PSO

This section describes an optimal approach based on the interval type-2 fuzzy logic system and PSO (IT2FLS-PSO). Each of the above parameters can be considered a particle in a population. Each particle will include the position and velocity moving in the search space. The PSO algorithm will stop if the position of the particle is optimal or when the number of loops is satisfied. IT2FLS-PSO is used for prediction purposes.

In this study, we use Gaussian functions to build the IT2 MFs for IT2FLS. It can be seen that the IT2 Gaussian membership function is characterized by parameters $m_j^i, \sigma_{j,1}^i, \sigma_{j,2}^i$. The original IT2FLS model was created for the study area using the training dataset. The antecedent and the consequent parameters of the initial model are not optimized. The PSO technique is then used to optimize the model by finding the best parameter values of antecedents and consequent parameters.

To increase model training effectiveness, the fuzzy c-means algorithm (FCM) is used to transfer samples into clusters, the initial parameters of IT2 Gaussian functions are the

centroids and standard deviation of the clusters. The clusters are used to generate the IF-THEN rules of the IT2FLS model. The best values for the antecedent and consequent parameters of the rules are obtained through the optimization iteration process.

The goal of this step is to find the optimal parameters for the IT2 Gaussian membership functions. When the original swarm was created, the initial position and velocity of each particle in the swarm were determined. Each particle will include position and velocity, which is limited to the search space $v_{min} \leq v_i^k \leq v_{max}$ and $vt_{min} \leq vt_i^k \leq vt_{max}$; in this case:

IF $v_i^k < v_{min}(v_i^k > v_{max})$ THEN $v_i^k = v_{min}(v_i^k = v_{max})$.
IF $vt_i^k < vt_{min}(vt_i^k > vt_{max})$ THEN $vt_i^k = vt_{min}(vt_i^k = vt_{max})$.

The optimization process is established by considering by the two parameters $P_{best}$ and $G_{best}$, wherein $P_{best}$ is the best solution of the particle at the present location and $G_{best}$ is the best solution of the population at the current location. These values will be updated based on the objective function value 7.

The training will be completed when the maximum number of iterations is reached, and when the optimization process is finished. The best position of the swarm is determined when the RMSE value is the smallest. Therefore, the values optimized for the antecedent and consequent parameters are determined for the IT2FLS-PSO model. The final model is then validated using validation data sets to confirm the accuracy and used for prediction purposes.

**Algorithm 4** IT2FLS optimization using PSO technique

**Input**: $X = (x_1, x_2, \ldots, x_n)$, number of rules $M$, number of IT2 Gaussian membership functions.
**Step 1**: Initialize parameters for the IT2 Gaussian MFs: Perform the FCM algorithm to initialize $m_j^i$ and using labelled data to initialize the standard deviation.
**Step 2**: Compute the lower and upper membership function of $x_j$:
$\mu_{\tilde{F}_j^i}(x_j) = [\underline{\mu}_{\tilde{F}_j^i}(x_j), \bar{\mu}_{\tilde{F}_j^i}(x_j)]$

**Step 3**: Compute the firing interval of the $i$th rule, where $*$ denotes the product operation.
$\underline{f}^i(x) = \underline{\mu}_{\tilde{F}_1^i}(x_1) * \underline{\mu}_{\tilde{F}_2^i}(x_2) * \ldots * \underline{\mu}_{\tilde{F}_n^i}(x_n)$

$\bar{f}^i(x) = \bar{\mu}_{\tilde{F}_1^i}(x_1) * \bar{\mu}_{\tilde{F}_2^i}(x_2) * \ldots * \bar{\mu}_{\tilde{F}_n^i}(x_n)$

**Step 4**: Compute the left most and right most output $[y_l^i, y_r^i]$ of the $i$th fuzzy rule.
**Step 5**: The crisp output for each output is calculated as follows: $\bar{y}_i = \frac{y_l^i + y_r^i}{2}$
**Step 6**: Compute RMSE: $RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^{n} (y_i - \bar{y}_i)^2}$
**Step 7**: If $(RMSE < \varepsilon)$ or $(Loop > Loop_{max})$ go to step Output.
**Step 8**: Implementing algorithm 1 and goto step 2.
**Output**: The parameters for IT2FLS.

The computational complexity of the proposed IT2FLS will also include the computational complexity of FCM in step 1, the computational complexity of PSO in step 8. The PSO algorithm adjusts the parameters of the IT2 Gaussian membership functions after each implementation of IT2FLS. The process is repeated until the optimal parameter for the fuzzy system is found. The parameters of the MFs will be used for the fuzzy system.

Because the PSO technique has smaller computational complexity than the genetic algorithm, evolutionary computation, by using this method it is possible to find the optimal parameters faster. However, their disadvantage is that finding the best parameters is contingent on the selection of the parameters of the PSO technique.

## 5. Experiments

The number of loops used to search for optimal parameters by the PSO technique is 1000. Because PSO does not guarantee convergence, experiments on PSO are conducted 30 times to find the best value. The capabilities of the proposed method are also compared to other methods including RF, KNN, SVM, multi-MLP, ANFIS and IT2FLS-FCM, in which IT2FLS-FCM is fuzzy type-2 interval system with initialization of parameters of IT2 Gaussian membership function using the FCM algorithm.

To evaluate the effectiveness of the proposed method, we measure the difference between the actual output and the desired output on the data sets labelled by Equation (7).

The resulting classification performance of the classification is evaluated by determining the True Positive Rate (TPR), and False Positive Rate (FPR) defined as follows:

$$\text{TPR} = \frac{\text{TP}}{\text{TP} + \text{FN}}, \quad \text{FPR} = \frac{\text{FP}}{\text{TN} + \text{FP}} \tag{9}$$

where TP is the number of correctly classified data, FN is the number of incorrectly misclassified data, FP is the number of incorrectly classified data and TN is the number of correctly misclassified data. The better the algorithm is, the higher the TPR value is and the smaller the FTR value is encountered.

### 5.1. Experiment 1

The first experimental data is the 'Forest Fires' dataset downloaded from the UCI Machine Learning Repository,[1] see Table 1. Data include 517 samples with 13 attributes. The paper uses five attributes including four inputs (temperature, relative humidity, wind speed, outside rain) and one output (burnt forest area) for prediction problem. This labelled sample data use 70% for training and 30% for validation.

In this experiment, IT2 Gaussian membership functions for four input and one output are constructed from the training data set. The result of the final model is checked by using the validation data set. The shape of the IT2 Gaussian membership functions is determined by parameters $m_j^i$, $\sigma_{j,1}^i$, $\sigma_{j,2}^i$. The input data before use will be mapped from the initial range into the range from 0 to 100. Output values range from 0.00 (ha) to 1100 (ha). This does not affect the prediction of the model but makes it easy to build interval type-2 fuzzy MFs. Once the optimal sets of three parameters have been determined, the IT2FLS can be used for predictive purposes.

The number of IT2 Gaussian membership functions that need to be designed is 20 for 4 inputs and 1 output. The initial population is initialized with 60 particles representing the shape of 20 IT2 Gaussian membership functions. This problem is modelled as a regression task, where the goal is to predict the burnt forest area.

**Table 1.** Experimental data.

| Data | Instances | Attributes | Used attributes |
|---|---|---|---|
| Forest Fires | 513 | 13 | 5 |
| Wine Quality-white | 4898 | 12 | 12 |
| Wine Quality-red | 1600 | 12 | 12 |

Table 2 is the result when predicted by RF, KNN, SVM, MPL, ANFIS, IT2FLS-FCM and IT2FLS-PSO models. The training result shows that the RMSE value of the IT2FLS-PSO model is 2.3834 for the training dataset and 2.4558 for the validation dataset, respectively. True positive rate is highest with 96.2803% for the training set and 95.7531% for the validation set. Meanwhile, the lowest false positive rate is only 0.0987% for the training set and 0.1152% for the validation set. In this experiment, KNN gave the lowest accuracy when the RMSE value was only 6.5372 for the training set and 5.2674 for the validation set. Moreover, the accuracy is only 91.5645% and 91.3299% for the training data set and the validation data set.

If the parameters of IT2FLS are initialized with FCM, the predicted results show that the accuracy is relatively high, second only to the IT2FLS-PSO method and higher than other methods.

Forest fires cause significant environmental damage while threatening human lives. The three major trends in handling this issue are the use of satellite data, infrared/smoke scanners and local sensors (e.g. meteorological). The sensor can instantly acquire these parameters. The advantage is that data can be collected in real-time and at a very low cost.

Predicting the extent of forest fires is a challenging task, and to improve accuracy, there is certainly a need for other information such as vegetation and soil. However, this research could open up opportunities for developing automated tools to assist the preparation for dealing with forest fires once they happen.

## 5.2. Experiment 2

The second experimental data is the 'Wine Quality' dataset downloaded from the UCI Machine Learning Repository[2] 1. Two datasets are included (wine-white data, wine-red data), related to red and white wine samples, from the north of Portugal. The goal is to model wine quality based on physicochemical tests.

Wine-white data includes 4898 samples with 12 attributes, and wine-red data includes 1599 samples with 12 attributes. The paper uses 12 attributes, including 11 inputs and 1 output for the prediction problem. The labelled data is used of 70% for training and 30% for validation.

The IT2FLS model should be designed to include 60 IT2 Gaussian membership functions for 12 inputs, each the input/output with 4 IT2 Gaussian membership functions. The initial population is initialized with 180 particles representing the shape of 60 IT2 Gaussian MFs.

**Table 2.** Burnt forest area prediction results on the 'Forest Fires' dataset.

| Model | Training dataset | | | Validation dataset | | |
|---|---|---|---|---|---|---|
| | RMSE | TPR (%) | FPR (%) | RMSE | TPR (%) | FPR (%) |
| RF | 3.8741 | 93.2232 | 0.1623 | 3.9873 | 92.2873 | 0.7642 |
| KNN | 6.5372 | 91.5645 | 0.5623 | 5.2674 | 91.3299 | 1.1736 |
| SVM | 4.1285 | 93.7416 | 0.2764 | 4.7648 | 92.6741 | 0.8763 |
| MLP | 3.2987 | 92.6572 | 0.1968 | 4.0841 | 93.0032 | 0.3652 |
| ANFIS | 3.6482 | 93.9853 | 0.1531 | 3.8774 | 93.7125 | 0.3309 |
| IT2FLS-FCM | 3.0198 | 94.9454 | 0.1543 | 3.6825 | 94.3362 | 0.2776 |
| IT2FLS-PSO | **0.3834** | **96.2803** | **0.0987** | **2.4558** | **95.7532** | **0.1152** |

Table 3 shows the results when predicted by RF, KNN, SVM, MPL, ANFIS, IT2FLS-FCM and IT2FLS-PSO models for wine-white data set. The training result shows that the RMSE value of the IT2FLS-PSO model is 1.6539 for the training dataset and 2.0762 for the validation dataset, respectively. True positive rate is highest with 98.2342% for the training set and 98.0377% for the validation set. Meanwhile, the lowest false positive rate is only 0.0784% for the training set and 0.1087% for the validation set.

Table 4 shows the results when predicted by RF, KNN, SVM, MPL, ANFIS, IT2FLS-FCM and IT2FLS-PSO models for wine-white data set. The training result shows that the RMSE value of the IT2FLS-PSO model is 1.9829 for the training dataset and 2.0764 for the validation dataset, respectively. True positive rate is highest with 98.0765% for the training set and 97.9882% for the validation set. Meanwhile, the lowest false positive rate is only 0.1018% for the training set and 0.2873% for the validation set.

Both models, the IT2FLS-FCM and the IT2FLS-PSO have better classification results than RF, KNN, SVM and MPL methods. In the second experiment, when the number of samples increased significantly, the accuracy of the proposed model was higher. More specifically, on the 'wine quality' dataset, the accuracy of IT2FLS-PSO model reached over 98% for training and over 97% for validation, while on the 'Forest Fires' dataset, this percentage is only above 96% and over 95%, respectively.

Through the above experiments, it is apparent that using the optimal technique to find the optimal parameters for IT2FLS can significantly increase the accuracy of IT2FLS.

## 6. Conclusion

The paper proposes the optimal model based on an interval type-2 fuzzy logic system with parameters defined by the PSO technique (IT2FLS-PSO). Experiments on data set downloaded from the UCI Machine Learning Repository. The experimental results in

**Table 3.** Wine quality prediction results on the wine-white dataset.

| Model | Training dataset | | | Validation dataset | | |
|---|---|---|---|---|---|---|
| | RMSE | TPR | FPR | RMSE | TPR | FPR |
| RF | 3.7761 | 94.6537 | 2.6741 | 3.6524 | 93.7682 | 2.6874 |
| KNN | 4.7838 | 93.8244 | 2.0264 | 4.7980 | 91.9841 | 3.0084 |
| SVM | 4.0874 | 94.1875 | 1.4673 | 4.5524 | 92.7664 | 1.3875 |
| MLP | 4.1853 | 93.9974 | 0.4562 | 4.8756 | 92.4983 | 0.4887 |
| ANFIS | 2.9882 | 95.7822 | 0.2571 | 3.5399 | 93.8211 | 0.4094 |
| IT2FLS-FCM | 2.7643 | 96.8753 | 0.2747 | 3.5242 | 95.3762 | 0.3781 |
| IT2FLS-PSO | **1.6539** | **98.2342** | **0.0784** | **2.0762** | **98.0377** | **0.1087** |

**Table 4.** Wine quality prediction results on the wine-red dataset.

| Model | Training dataset | | | Validation dataset | | |
|---|---|---|---|---|---|---|
| | RMSE | TPR | FPR | RMSE | TPR | FPR |
| RF | 3.1764 | 93.4877 | 2.3885 | 3.5761 | 93.2771 | 2.8779 |
| KNN | 4.3765 | 93.1875 | 2.7849 | 4.7563 | 92.0043 | 3.1642 |
| SVM | 4.6982 | 92.3885 | 1.9842 | 4.5983 | 92.2875 | 2.0063 |
| MLP | 4.4872 | 92.3872 | 1.4988 | 4.1758 | 92.1874 | 1.5629 |
| ANFIS | 3.0852 | 95.4631 | 0.8739 | 3.3789 | 93.8958 | 0.8805 |
| IT2FLS-FCM | 2.8728 | 95.9983 | 0.3984 | 3.4875 | 95.2765 | 0.6724 |
| IT2FLS-PSO | **1.9829** | **98.0765** | **0.1018** | **2.0764** | **97.9882** | **0.2873** |

this paper show that the IT2FLS-PSO model achieved better results than those produced by using the methods of RF, KNN, SVM, MLP, ANFIS and IT2FLS-FCM.

Both hybrid models, IT2FCM-FCM and IT2FLS-PSO, produced better results than RF, KNN, SVM, MLP, ANFIS models in most cases with the highest accuracy reaching over 98% for T2FLS-PSO and over 97% for IT2FLS-FCM.

Prediction problems are challenging tasks with considerable constrains by input data, predictive models, as well as parameters for each model. The hybridization model between IT2FLS and PSO technique has been shown to have enormous potential for a wide range of prediction application. This paper is the basis for the authors to experiment on many different types of membership functions for each data set.

In the future, we will develop hybrid predictive systems, combing fuzzy systems and deep learning.

## Notes

1. https://archive.ics.uci.edu/
2. https://archive.ics.uci.edu/

## Acknowledgments

## Disclosure statement

No potential conflict of interest was reported by the author(s).

## Funding

## Notes on contributors

*Dinh Sinh Mai* is a lecturer at the Institute of Techniques for Special Engineering and PhD student in fundamentals of mathematics for informatics at Department of Information System, Le Quy Don Technical University (LQDTU), Hanoi, Vietnam. He received the BS (2009) and MS degrees (2013) in GeoInformatics and Computer Science from LQDTU. His research interests are fuzzy logic, pattern recognition, image processing and GeoInformatics. Email: *maidinhsinh@gmail.com*.

*Trong Hop Dang* is a lecturer at Hanoi University of Industry, Vietnam. He received his PhD in 2019 at LQDTU. His main research interests are fuzzy clustering, pattern recognition and hybrid fuzzy systems. Email: *dangtronghop@gmail.com*.

*Long Thanh Ngo* received the MSc, PhD degrees from LQDTU, in 2003 and 2009, respectively. He is currently an associate professor with the Department of Information

Technology and Institute of Simulation Technology, LQDTU, Hanoi, Vietnam. His current research interests include computational intelligence, type-2 fuzzy logic, pattern recognition and image processing. Email: *ngotlong@gmail.com*.

## ORCID

*Dinh Sinh Mai* http://orcid.org/0000-0001-7284-5713
*Long Thanh Ngo* http://orcid.org/0000-0002-3856-3154

## References

Abbas, K., Saeid, N., Doug, C., & Dipti, S. (2012). Interval type-2 fuzzy logic systems for load forecasting: A comparative study. *IEEE Transactions on Power Systems*, *27*(3), 1274–1282. doi: https://doi.org/10.1109/TPWRS.2011.2181981

Alreshoodi, M., Adeyemi-Ejeye, A. O., Woods, J., Walker, S. D., & Pokhrel, J. (2016). Interval type-2 fuzzy logic quality prediction model for wireless 4kUHD H.265-coded video streaming. *2016 IEEE International Conference on Consumer Electronics* (pp. 407–408).

Amit, K. S., Megha, Y., Sandeep, K., & Pranab, K. M. (2020). Veracity handling and instance reduction in big data using interval type-2 fuzzy sets. *Engineering Applications of Artificial Intelligence*, *88*, 103315. doi: https://doi.org/10.1016/j.engappai.2019.103315

Ayad, A. M., Fendy, S., Matthew, A. G., & Sreenatha, G. A. (2019). An intelligent control of an inverted pendulum based on an adaptive interval type-2 fuzzy inference system. *FUZZ-IEEE*.

Chao, C., Robert, J., Jamie, T., & Jonathan, M. G. (2016). An extended ANFIS architecture and its learning properties for type-1 and interval type-2 models. *FUZZ-IEEE* (pp. 602–609).

Claudia, I.G., Patricia, M., Juan, R. C., Oscar, C., & Olivia, M. (2016). Optimization of interval type-2 fuzzy systems for image edge detection. *Applied Soft Computing*, *47*, 631–643. doi: https://doi.org/10.1016/j.asoc.2014.12.010

Das, A. K., Subramanian, K., & Sundaram, S. (2015). An evolving interval type-2 neuro-fuzzy inference system and its meta-cognitive sequential learning algorithm. *IEEE Transactions on Fuzzy Systems*, *23*(6), 2080–2093. doi: https://doi.org/10.1109/TFUZZ.2015.2403793

Das, K., Suresh, S., & Sundararajan, N. (2016). A fully tuned sequential interval type-2 fuzzy inference system for motor-imagery task classification. *IEEE International Conference on Fuzzy Systems* (pp. 751–758).

Dzung, D. N., Long, T. N., Long, P. T., & Pedrycz, W. (2015). Towards hybrid clustering approach to data classification: Multiple kernel based-interval-valued fuzzy c-means algorithms. *Fuzzy Sets and Systems*, *279*, 17–39. doi: https://doi.org/10.1016/j.fss.2015.01.020

Eduardo, R., Patricia, M., & German, P. A. (2019). Hybrid model based on neural networks, type-1 and type-2 fuzzy systems for 2-lead cardiac arrhythmia classification. *Expert Systems With Applications*, *126*, 295–307. doi: https://doi.org/10.1016/j.eswa.2019.02.035

Emanuel, O. R., & Patricia, M. (2019). A hybrid design of shadowed type-2 fuzzy inference systems applied in diagnosis problems. *Engineering Applications of Artificial Intelligence*, *86*, 43–55. doi: https://doi.org/10.1016/j.engappai.2019.08.017

Emanuel, O., Patricia, M., & Oscar, C. (2020). Comparative study of interval type-2 and general type-2 fuzzy systems in medical diagnosis. *Information Sciences*, *525*, 37–53. doi: https://doi.org/10.1016/j.ins.2020.03.059

Emanuel, O. R., Patricia, M., & Oscar, C. (2019). Relevance of polynomial order in Takagi–Sugeno fuzzy inference systems applied in diagnosis problems. *FUZZ-IEEE*.

Fernando, G., Patricia, M., Fevrier, V., & Juan, R. C. (2016). Optimization with genetic algorithm and particle swarm optimization of type-2 fuzzy integrator for ensemble neural network in time series. *FUZZ-IEEE*.

Frumen, O., Fevrier, V., Oscar, C., Claudia, I. G., Gabriela, M., & Patricia, M. (2017). Ant colony optimization with dynamic parameter adaptation based on interval type-2 fuzzy logic systems. *Applied Soft Computing*, *53*, 74–87. doi: https://doi.org/10.1016/j.asoc.2016.12.015

Juan, E. M., Mauricio, A. S., Olivia, M., Antonio, R. D., Oscar, C., Patricia, M., & Juan, R. C. (2020). Design of an interval type-2 fuzzy model with justifiable uncertainty. *Information Sciences*, *513*, 206–221. doi: https://doi.org/10.1016/j.ins.2019.10.042

Kennedy, J., & Eberhart, R. (1995). Particle swarm optimization. *IEEE international conference on neural networks* (pp. 1942–1948).

Li, J., Yang, L., Fu, X., Chao, F., & Qu, Y. (2018). Interval type-2 TSK+ fuzzy inference system. *FUZZ-IEEE*.

Liang, Q., & Mendel, J. (2000). Interval type-2 fuzzy logic systems: Theory and design. *IEEE Transactions on Fuzzy Systems*, *8*(5), 535–550. doi: https://doi.org/10.1109/91.873577

Lin, Y. Y., Liao, S. H., Chang, J. Y., & Lin, C. T. (2014). Simplified interval type-2 fuzzy neural networks. *IEEE Transactions on Neural Networks and Learning Systems*, *25*(5), 959–969. doi: https://doi.org/10.1109/TNNLS.2013.2284603

Long, T. N., Mai, D. S., & Nguyen, M. U. (2012). GPU-based acceleration of interval type-2 fuzzy c-means clustering for satellite imagery land-cover classification. *IEEE international conference on intelligent systems design and applications* (pp. 992–997).

Long, T. N., Mai, D. S., & Pedrycz, W. (2015). Semi-supervising interval type-2 fuzzy c-means clustering with spatial information for multi-spectral satellite image classification and change detection. *Computers and Geosciences*, *83*, 1–16. doi: https://doi.org/10.1016/j.cageo.2015.06.011

Lu, Q., Shi, P., H. K. Lam, & Zhao, Y. (2015). Interval type-2 fuzzy model predictive control of nonlinear networked control systems. *IEEE Transactions on Fuzzy Systems*, *23*(6), 2317–2328. doi: https://doi.org/10.1109/TFUZZ.2015.2417975

Maciel, L., & Ballini, R. (2019). A fuzzy inference system modeling approach for interval-valued symbolic data forecasting. *Knowledge-Based Systems*, *164*, 139–149. doi: https://doi.org/10.1016/j.knosys.2018.10.033

Mai, D. S. (2018). Satellite image classification based spatial-spectral fuzzy clustering algorithm. In *The 10th Asian conference on intelligent information and database systems* (pp. 505–518). Springer.

Mai, D. S, & Long, T. N. (2018). The semi-supervised method with spatial weights based possibilistic fuzzy C-means clustering for Land-cover classification. *The 6th NAFOSTED conference on information and computer science* (pp. 406–411), IEEE.

Mai, D. S., & Long, T. N. (2015). Semi-supervised fuzzy c-means clustering for change detection from the multispectral satellite image. *FUZZ-IEEE* (pp. 1–8).

Mai, D. S., & Long, T. N. (2019). General semi-supervised possibilistic fuzzy c-Means clustering for land-cover classification. In *The 11th IEEE international conference on knowledge and systems engineering* (pp. 1–6).

Mai, D. S, Long, T. N., & Trinh, L. H. (2018b). Spatial-spectral fuzzy k-means clustering for remote sensing image segmentation. *Vietnam Journal of Science and Technology*, *56*(2), 257–272. doi: https://doi.org/10.15625/2525-2518/56/2/10785

Mai, D. S., Long, T. N., & Trinh, L. H. (2018a). Advanced semi-supervised possibilistic fuzzy c-means clustering using spatial-spectral distance for land-cover classification. *The IEEE international conference on systems, man, and cybernetics* (pp. 4375–4380).

Mai, D. S., Ngo, L. T., & Trinh, L. H. (2020). Approach the interval type-2 fuzzy system and PSO technique in Landcover classification. In N. Nguyen, K. Jearanaitanakij, A. Selamat, B. Trawiński, & S. Chittayasothorn (Eds.), *Intelligent information and database systems*. ACIIDS 2020. Lecture Notes in Computer Science, 12034. Springer.

Mendel, J. M. (2007). Type-2 fuzzy sets and systems: An overview. *IEEE Computational Intelligence Magazine*, *2*(1), 20–29. doi: https://doi.org/10.1109/MCI.2007.380672

Mendel, J. M. (2017). *Uncertain rule-based fuzzy logic systems: Introduction and new directions* (2nd ed). Springer.

Mendel, J. M., John, R. I., & Liu, F. (2006). Interval type-2 fuzzy logic systems made simple. *IEEE Transactions on Fuzzy Systems*, *14*(6), 808–821. doi: https://doi.org/10.1109/TFUZZ.2006.879986

Mendel, J., & Liu, X. (2013). Simplified interval type-2 fuzzy logic systems. *IEEE Transactions on Fuzzy Systems*, *21*(6), 1056–1069. doi: https://doi.org/10.1109/TFUZZ.2013.2241771

Nguyen, T., & Saeid, N. (2016). Modified AHP for gene selection and cancer classification using type-2 fuzzy logic. *IEEE Transactions on Fuzzy Systems*, *24*(2), 273–287. doi: https://doi.org/10.1109/TFUZZ.2015.2453153

Oscar, C., & Patricia, M. (2012). A review on the design and optimization of interval type-2 fuzzy controllers. *Applied Soft Computing*, *12*(4), 1267–1278. doi: https://doi.org/10.1016/j.asoc.2011.12.010

Oscar, C., & Patricia, P. (2019). An approach for optimization of intuitionistic and type-2 fuzzy systems in pattern recognition applications. *FUZZ-IEEE*, 19046225.

Panella, M., & Rizzi, A. (2003). A Neuro-fuzzy system for the prediction of the vehicle traffic flow. In V. Di Gesú, F. Masulli, & A. Petrosino (Eds.), *Fuzzy logic and applications*. WILF 2003. Lecture Notes in Computer Science, 2955. Springer.

Patricia, M., & Oscar, C. (2014). A review on type-2 fuzzy logic applications in clustering, classification and pattern recognition. *Applied Soft Computing*, *21*, 568–577. doi: https://doi.org/10.1016/j.asoc.2014.04.017

Shihabudheen, K. V., Mahesh, M., & Pillai, G. N. (2018). Particle swarm optimization based extreme learning neuro-fuzzy system for regression and classification. *Expert Systems with Applications*, *92*, 474–484. doi: https://doi.org/10.1016/j.eswa.2017.09.037

Thanh, N., & Saeid, N. (2015). Medical data classification using interval type-2 fuzzy logic system and wavelets. *Applied Soft Computing*, *30*, 812–822. doi: https://doi.org/10.1016/j.asoc.2015.02.016

Trinh, L. H., & Mai, D. S. (2019). Classification of remote sensing imagery based on density and fuzzy c-means algorithm. *International Journal of Fuzzy System Applications*, *8*(2), 1–15. https://doi.org/10.4018/IJFSA.2019040101

Wu, D., & Mendel, J. M. (2019). Recommendations on designing practical interval type-2 fuzzy systems. *Engineering Applications of Artificial Intelligence*, *85*, 182–193. doi: https://doi.org/10.1016/j.engappai.2019.06.012