IT2-GSETSK: An Evolving Interval Type-II TSK Fuzzy Neural System for Online Modeling of Noisy Data

Mohammad Ashrafi[a,c*], Dilip K. Prasad[b], and Chai Quek[a]

[a] School of Computer Science and Engineering, Computational Intelligence Laboratory, Nanyang Technological University, 50 Nanyang Avenue, Singapore 639798.

[b] Department of Computer Science, UiT The Arctic University of Norway, Tromsø, Norway 9037.

[c] Current affiliation: iTrust Centre for Research in Cyber Security, Singapore University of Technology and Design (SUTD), 8 Somapah Road, Singapore 487372.

*Corresponding author

E-mail addresses: mohammad021@e.ntu.edu.sg, mashrafi.eng@gmail.com (M.Ashrafi), dilip.prasad@uit.no (D.K. Prasad); ashcquek@ntu.edu.sg (C.Quek)

ABSTRACT

As a core part of a fuzzy neural system, the rule base antecedents and consequents may carry uncertainties because they are trained using noisy data. So, handling the uncertain rule base is an important need in some specific problems to have a better data modeling. As a solution, Interval Type-II (IT2) version of GSETSK (Generic Self-Evolving Takagi-Sugeno-Kang), namely IT2-GSETSK, is presented in this paper. This solution uses IT2 membership functions for handling uncertainties, plus having Type-I (GSETSK) capabilities. In this way IT2-GSETSK is a fully-online model able to handle data streams and cope with time-variant data. It also provides up-to-date, relevant and compact rule base that is easily interpretable by human. The IT2-GSETSK is tested over several applications including medical, environmental and financial predictions, which show satisfactory performance of IT2-GSETSK. Moreover, it is observed that while GSETSK performs well enough for dynamic problems with less noise, noisy non-dynamic problems benefit significantly from IT2-GSETSK.

Keywords: Type-II, Self-Evolving, GSETSK, Neuro-fuzzy, Noisy Data, Non-stationary

1. INTRODUCTION

Type-1 fuzzy sets and systems assume no uncertainty associated with data, meaning that there is no system unpredictability and that reduction from probability to determinism is possible. However, systems are uncertain in reality. So, fuzzy sets that handle uncertainties were invented, among which Type-2 fuzzy sets [1] are the most well-developed and well-studied. Specifically, Interval Type-2 (IT2) fuzzy sets [2-4] model the uncertainty by getting the footprint of uncertainty (FOU). An IT2 fuzzy set can be imagined as two Type-1 fuzzy sets, one bounding the upper membership function and the other bounding the lower membership function. In a Takagi-Sugeno-Kang (TSK) fuzzy system, IT2 fuzzy sets handle the uncertainties related to: 1- the cluster center and extending to the antecedent part, and 2- the consequent parameters.

Interval Type-2 Fuzzy Neural networks (IT2FNN) have been widely used since their invention [5]. As a review in [6] shows, there are more than 6000 publications during 2009-2017 using IT2FNN for chaotic time series prediction, which show an increasing trend with applications in different areas. However as highlighted in [6], application of IT2FNN in online and evolving modeling is still in the nascent stage and needs more research.

Evolving FNNs [7, 8] have specific characteristics which are desirable for real-life applications. In non-stationary or evolving environment [9], the distribution of the data related to the behavior of a specific phenomenon changes gradually to abrupt and beyond its historical bounds. Today, the advancement in the sensor technology and related sciences provides better tracking of the phenomena in the form of data streams. With online structure and parameter learning, evolving FNNs can handle data streams in real-time based on the last data seen. This eventually leads to online FNN with low computational complexity. Evolving IT2FNNs portend to have the additional advantage of handling uncertainties comparing to evolving FNNs.

Studies introduced as evolving IT2FNNs using 1^{st}-order TSK [10] in the literature includes T2SONFS [11], SEIT2FNN [12], its recurrent version RSEIT2FNN [13], IT2FNN-SVR [14], MRIT2NFS [15], and TSCIT2FNN [16]. However, none of these models are fully-online as all of them need data’s upper and lower bounds for data normalization, e.g. in the range of [-1, 1]. The other problem with these models is the absence of rule management mechanisms. Consequently, the shortage of rule pruning module in these models makes them ever-growing and unsuitable to handle non-stationary problems. Also, the absence of rule merging mechanism puts the model at the risk of overfitting with hardly-interpretable rule-base.

It was tried to fix the problem of rule management in the later studies. In one study, RIT2NFS-WB [17], a fuzzy set merging module was proposed to reduce highly-overlapping fuzzy sets. In another study, SRIT2NFIS [18], rule pruning was carried out for classification problems in non-stationary environment. Still, neither model is fully online as both need the data's lower and upper bounds.

Recently, two advanced evolving IT2FNNs, namely eT2RFNN [19] and eT2Class [20], were proposed for regression and classification tasks respectively. The rule merging and pruning mechanisms in both models empower the models for handling the non-stationarity of the data. Some other studies have used metacognitive theory and IT2 fuzzy sets, which provide functions similar to evolving IT2FNNs. McIT2FIS [21] is an example model for regression problems with rule pruning module. But it still needs data normalization. Also, RIVMcSFNN [22] and eT2ELM [23] applied to regression and classification problems respectively. Both models are equipped with rule pruning modules and the former one is equipped with rule merging module as well. To the best of our knowledge, SC-IT2FNN [24], and SOIT2FNN [25] are the other IT2-evolving neuro-fuzzy models (NFMs) with rule growing and pruning mechanism.

Through this paper, we contribute a state-of-the-art evolving IT2FNN model, namely IT2-GSETSK. The proposed IT2-GSETSK can cope with noisy data in non-stationary regression problems and provides an interpretable rule-base. It has the following specific features:

1- Starting from scratch, IT2-GSETSK has online structure and parameter learning, while handling the data stream in an interleaving test-and-train manner. This means that we update the antecedent of the rule base using the current data sample, predict the output, and then tune the consequent part of the rule base using the predicted output.

2- The Multidimensional-Scaling Growing Clustering (MSGC) method used in IT2-GSETSK does not need any prior-knowledge about the data. MSGC projects the real inputs on separate 1-D input spaces. This makes IT2-GSETSK a fully-online and unbounded, without requiring data normalization, which eventually allows better handling of non-stationary problems.

3- The rule pruning (i.e., unlearning) module of IT2-GSETSK helps the structure learning to forget the irrelevant rules, allowing it to better cope with the last changes in the process.

4- The rule merging module manages the rule-base to have non-overlapping and distinct rules, which eventually reduces the risk of overfitting and provides more interpretable rule-base.

5- IT2 fuzzy sets can handle uncertainties with the cluster centers and cluster extent, expansion in the input space, and expansion in surface function relating to consequent parameters.

Real-life phenomena are often complex and challenging to model. Some phenomena occur for a short duration of time, some are fast-changing, and there are some cases where no historical data is available. Automatic ventilation for a patient, volatile stock market prediction, and flood forecasting in a newly-gauged watershed are examples of such phenomena. Modeling such phenomena needs a fast-reaction mechanism which can capture and model the behavior of the system quickly. In addition, noisy data needs to be handled to not affect the model. IT2-GSETSK is capable to handle the challenges mentioned above, as is demonstrated by our results on such modeling problems.

The layout of our paper is as follows. The architecture of IT2-GSETSK, and its structure and parameter learning approach are presented in Section 2. The experimental results are presented in Section 3, followed by a discussion in Section 0 and conclusion in Section 5.

2. IT2-GSETSK MODEL

Similar to GSETSK [26, 27], IT2-GSETSK has a six-layer structure (Fig. 1) receiving the inputs at Layer I and delivering the outputs at Layer VI. The core part of the model is the rule base at Layer III which evaluates the IF statements with antecedents from Layer II and relates them to the rule consequents in Layer V. In IT2-GSETSK, IT2 fuzzy sets are used in each rule (as in Eq. (1)) to handle the uncertainties

$$R^k\colon IF\ x_1\ is\ \tilde{A}_1^k\ \ AND \cdots AND\ x_n\ is\ \widetilde{A}_n^k,\ \text{THEN}\ y = \tilde{b}_0^k + \sum_{i=1}^{n} \tilde{b}_i^k x_i\,,\quad k = 1, \dots, K(t), \qquad (1)$$

where R^k is the *k(th)* rule, ($k = 1, \dots, K(t)$), x_i is data on the specific input dimension *i*, ($i = 1, \dots, n$), $\tilde{A}_i^k$ is IT2 fuzzy set of the *k(th)* rule in the input dimension *i*, and $\tilde{b}_i^k$ is the interval consequent parameter that relates to input dimension *i* of *k(th)* rule.

IT2-GSETSK has a self-evolving structure, which is alive through the lifetime of the model such that it can construct or deconstruct its structure to cope with the complexities of the problem. This is handled in IT2-GSETSK through specific mechanisms for structure learning and unlearning and with a localized version of recursive linear least-squares (RLS) [26, 27] for tuning the consequent parameters. The model operates in an interleaved test-and-train mode upon arrival of each data to support the online self-evolution of the model. The test-and-train begins after model initialization using the 1st input-output data sample, which is responsible for generating the 1st cluster. From the 2nd sample onwards, the interleaved test-and-train procedure is employed as follows:

1. Update the antecedent part of the rule base based on the input of the current data sample.
2. Predict the output by using the input of the current data sample.
3. Tune the consequent part of the rule base using the output of the most recent data sample.

In the following sections, the architecture of IT2-GSETSK and the structure learning and unlearning are explained, followed by interpretability of IT2-GSETSK.

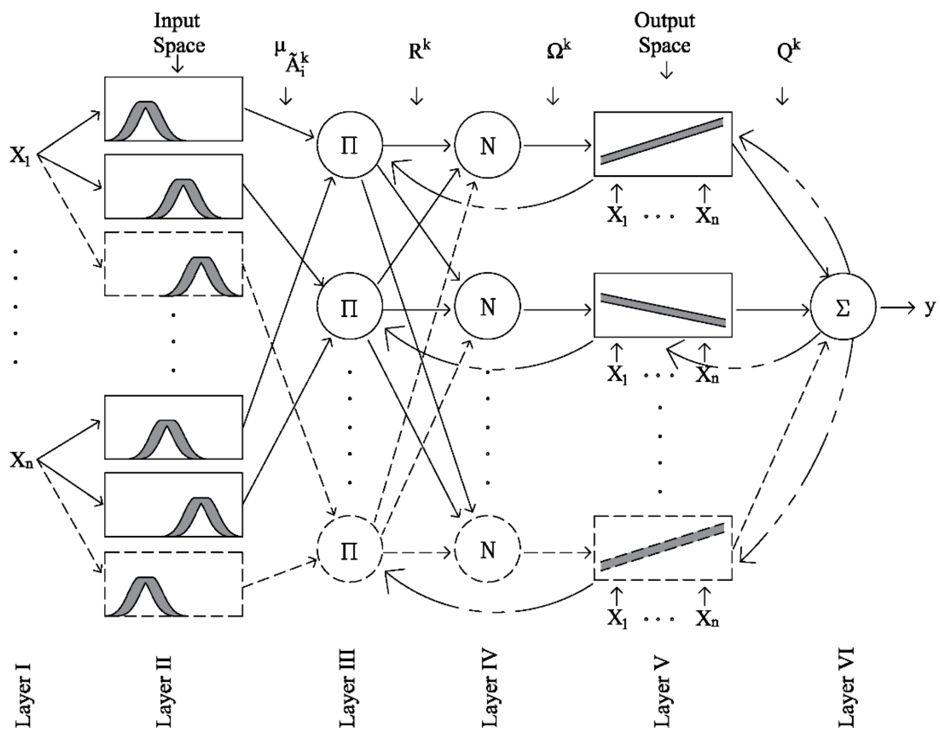

Fig. 1. IT2-GSETSK architecture.

2.1. IT2-GSETSK Architecture

IT2-GSETSK has a 6-layer structure (Fig. 1), with feedback connections for model unlearning from Layer VI to Layer III via Layer V. The layers and their functions are as follows.

Layer I (input layer): The input layer makes singletons of the input values at each input dimension $(x_i,\ i = 1, \ldots, n)$ and transfers them to the next layer. As mentioned before, there is no need of data normalization to unify different input dimensions in IT2-GSETSK. Instead, real inputs are projected on 1-D input spaces related to each specific input variable. This is an advantage over most of the FNNs. This is accomplished in IT2-GSETSK through a clustering algorithm discussed in Section 2.2.

Layer II (fuzzification layer): Each node in this layer is an IT2 fuzzy set ($\tilde{A}_i^k$) with lower and upper bounds representing the footprint of uncertainty (FOU). Gaussian membership functions (GMF) are used to represent the lower and upper bounds of fuzzy sets. When the model operates, an interval degree of membership ($\mu_{\tilde{A}_i^k}$) is computed by evaluating the input value against each IT2 fuzzy set:

$$\mu_{\tilde{A}_i^k} = \left[\underline{\mu}_{\tilde{A}_i^k}, \overline{\mu}_{\tilde{A}_i^k}\right], \tag{2}$$

where $\underline{\mu}_{\tilde{A}_i^k}$ and $\bar{\mu}_{\tilde{A}_i^k}$ are the degrees of membership for lower and upper bounds respectively.

There are two types of IT2 fuzzy sets used in this research (Fig. 2b1,c1): IT2 fuzzy set with uncertain mean (IT2-UM) and IT2 fuzzy set with uncertain standard deviation (IT2-USTD). These two types of IT2-GSETSK provides us greater opportunity to capture uncertainties as they cover different areas on the input space. As illustrated in Eq. (3), IT2-UM is constructed using two different centers (c_{i1}^k and c_{i2}^k) and a fixed standard deviation (STD). The degree of membership of a datum can be computed based on the distance of the datum from the two centers as in Eq. (4-5).

$$\mu_{\tilde{A}_i^k} = \exp\left(-\frac{1}{2}\left(\frac{x_i - c_i^k}{\sigma_i^k}\right)^2\right) \equiv N(c_i^k, \sigma_i^k; x_i),\ \ c_i^k \in [c_{i1}^k, c_{i2}^k], \tag{3}$$

where $\tilde{A}_i^k$ is the IT2 fuzzy set of *k*th rule on input dimension x_i, c_i^k is the center of $\tilde{A}_i^k$, and σ_i^k is the STD of $\tilde{A}_i^k$

$$\bar{\mu}_{\tilde{A}_i^k}(x_i) = \begin{cases} N(c_{i1}^k, \sigma_i^k; x_i), & x_i < c_{i1}^k \\ 1, & c_{i1}^k \le x_i \le c_{i2}^k \\ N(c_{i2}^k, \sigma_i^k; x_i), & x_i > c_{i2}^k \end{cases}, \tag{4}$$

$$\underline{\mu}_{\tilde{A}_i^k}(x_i) = \begin{cases} N(c_{i2}^k, \sigma_i^k; x_i), & x_i \le \left(c_{i1}^k + c_{i2}^k\right)/2 \\ N(c_{i1}^k, \sigma_i^k; x_i), & x_i > \left(c_{i1}^k + c_{i2}^k\right)/2 \end{cases}, \tag{5}$$

The IT2-USTD uses different STDs (σ_{i1}^k and σ_{i2}^k) and a constant center (c_i^k) as in Eq. (6). Then, each upper and lower degree of membership can be computed using Eqs. (7, 8).

$$\mu_{\tilde{A}_i^k} = \exp\left[-\frac{1}{2}\left(\frac{x_i - c_i^k}{\sigma_i^k}\right)^2\right] \equiv N(c_i^k, \sigma_i^k; x_i), \qquad \sigma_i^k \in \left[\sigma_{i1}^k, \sigma_{i2}^k\right], \tag{6}$$

$$\bar{\mu}_{\tilde{A}_i^k}(x_i) = N(c_i^k, \sigma_{i2}^k; x_i), \tag{7}$$

$$\underline{\mu}_{\tilde{A}_i^k}(x_i) = N(c_i^k, \sigma_{i1}^k; x_i), \tag{8}$$

Layer III (rule layer): This layer carries the information about the range of influence of each rule at the model output. The interval firing strength of each rule (r^k) is computed in each rule node of this layer (Eq. (9)) which defines the upper and lower degree of activation of a rule as in Eq. (4).

$$r^k = \left[\underline{r}^k, \bar{r}^k\right], \tag{9}$$

$$\bar{r}^k = \prod_{i=1}^{n} \bar{\mu}_{\tilde{A}_i^k}, \text{ and } \underline{r}^k = \prod_{i=1}^{n} \underline{\mu}_{\tilde{A}_i^k}, \tag{10}$$

The Π (minimization) operation ensures lowering the degree of uncertainty as the lower and upper firing strengths are related to the level of coincidence of all the predictors (input spaces).

Layer IV (normalization layer): The output of each node in this layer (Ω^k) is an interval (Eq. (11)) which shows the upper and lower degrees of the influence of each rule on the whole model. The upper and lower normalized firing strengths ($\bar{\omega}^k$ and $\underline{\omega}^k$, respectively) are computed as in Eq. (12).

$$\Omega^k = \left[\underline{\omega}^k, \bar{\omega}^k\right], \tag{11}$$

$$\bar{\omega}^k = \bar{r}^k / \sum_{k=1}^{K(t)} \bar{r}^k, \quad \text{and} \quad \underline{\omega}^k = \underline{r}^k / \sum_{k=1}^{K(t)} \underline{r}^k, \tag{12}$$

Layer V (consequent Layer): Since IT2-GSETSK uses 1st-order TSK [10] neuro-fuzzy model, the consequent part is in the form of a linear equation. Each node describes the output space related to the antecedent part (cluster) of that rule as an interval ($F^k(X)$ in Eq. (13)) with lower and upper bounds as in Eq. (14).

$$F^k(X) = \left[\underline{f}^k(X), \bar{f}^k(X)\right], \tag{13}$$

$$\bar{f}^k(X) = \bar{b}_0^k + \bar{b}_1^k x_1 + \cdots + \bar{b}_n^k x_n, \qquad \underline{f}^k(X) = \underline{b}_0^k + \underline{b}_1^k x_1 + \cdots + \underline{b}_n^k x_n, \tag{14}$$

where the consequent parameters in Eq. (14) can be arranged in two matrices (as in Eq. (15)) that are later used for model parameter learning [26].

$$\bar{B}^k = [\bar{b}_0^k,\ \bar{b}_1^k,\ \ldots,\ \bar{b}_n^k], \quad \text{and} \quad \underline{B}^k = \left[\underline{b}_0^k,\ \underline{b}_1^k,\ \cdots,\ \underline{b}_n^k\right], \tag{15}$$

Covering an upper and lower degree of uncertainty, each input space partition has respective upper and lower planes in the output space. The effect of each rule is computed by multiplying the normalized firing strength with the output of the linear equation in the output space. This forms two interval output ($\bar{q}^k$ and $\underline{q}^k$) as in Eq. (16) and their average is computed for the final outcome (Q^k) of a rule as in Eq. (17).

$$\bar{q}^k = \bar{\omega}^k \bar{f}^k(X) \quad \text{and} \quad \underline{q}^k = \underline{\omega}^k \underline{f}^k(X), \tag{16}$$

$$Q^k = (\underline{q}^k + \bar{q}^k)/2, \tag{17}$$

Layer VI (summation layer): In this layer, outcomes of all rules are summed up to generate the final result (y) as in Eq. (18).

$$y = \sum_{k=1}^{K(t)} Q^k, \tag{18}$$

2.2. IT-GSETSK Structure Learning

The structure of IT2-GSETSK is continuously updated based on the problem complexities in the model's lifetime. There are two learning and unlearning mechanisms responsible to construct and deconstruct the model, which we discuss in the following.

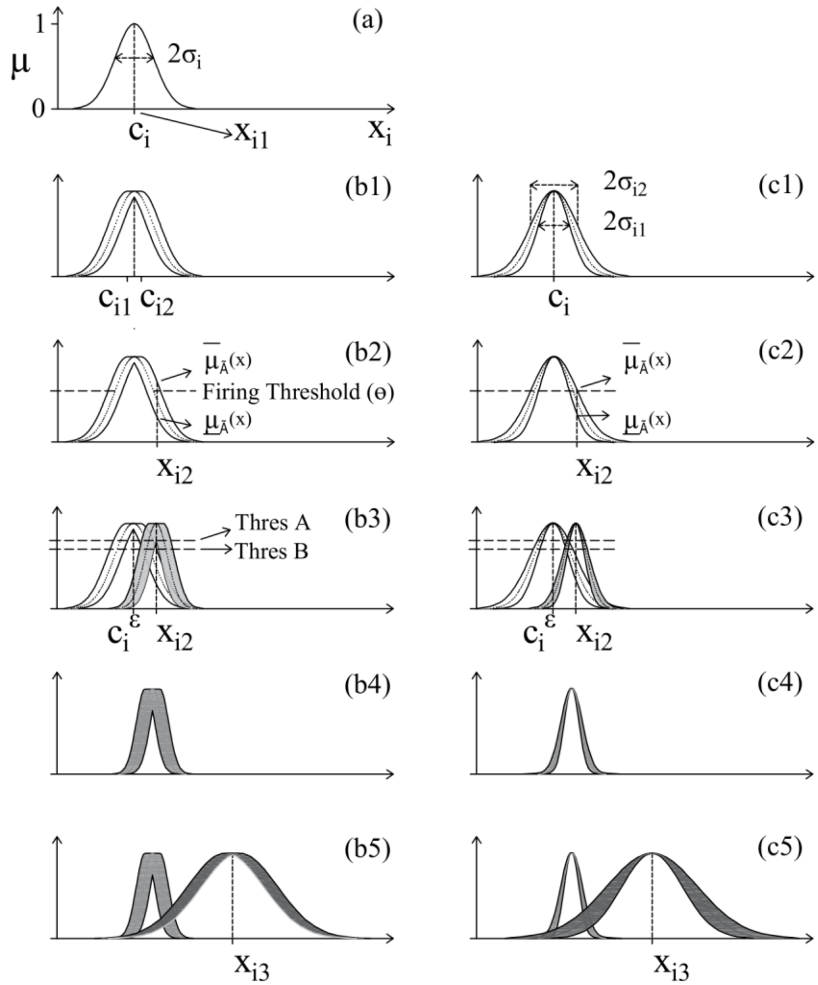

Fig. 2. MSGC mechanism from scratch dealing with 3 arrival data (xi1 ,xi2, xi3) at different steps from IT1 (a) to IT2-UM (b1-b5) and IT2-USTD (c1-c5) GMFs; a) creation of the 1st IT1 GMF; b1,c1) conversion to IT2 GMFs; b2,c2) evaluation of degree of activation of the arrived data with closest GMF; b3,c3) similarity measurement; b4,c4) merging of GMFs; b5,c5) generation of new GMF without merging.

2.2.1. Learning Algorithm

The learning mechanism in IT2-GSESK is designed to make the model fully-online, which needs no prior-knowledge of data. Also, it creates non-overlapping clusters which cover the entire cluster space very well. This is supported by Multidimensional-Scaling Growing Clustering (MSGC) [26, 27] used for partitioning the input space. The steps of MSGC are described below and illustrated using three example samples in Fig. 2. Initially, there is no rule in IT2-GSETSK. During the online execution of the model, for each newly arrived set of input data (such as x_{i1} in Fig. 2a), the model first creates a temporary rule k with temporary IT1 GMFs for different input dimensions. This frees the model from normalization which needs the prior knowledge of data's upper and lower bounds. There is one GMF in each specific dimension *i* (Fig. 2a) with center (c_i^k) and STD (σ_i^k) as in Eq. (19)

$$c_i^k = x_i, \qquad \sigma_i^k = \upsilon, \tag{19}$$

where υ is a predefined value for the 1st cluster. For the 1st cluster, υ is defined based on expert knowledge. The temporary GMFs in the 1st cluster are of the type IT1. They need to be converted to IT2 to cope with the uncertainties. This is done by multiplying the center or STD of the temporary GMF (Eq. (19)) by the degree of uncertainty ($\alpha \in (0.05,0.95)$) as in Eq. (20). Hence, the center and the STD of the new GMF (Fig. 2b1, c1) are computed as in Eq. (21) for IT2-UM. Eq. (22) shows the computation of the center and the STD of new GMF for IT2-USTD.

$$\vartheta_i = \alpha c_i^k, \qquad \zeta = \alpha \sigma_i^k, \tag{20}$$

$$\left[c_{i1}^k, c_{i2}^k\right] = \left[c_i^k - \vartheta_i, c_i^k + \vartheta_i\right], \qquad \sigma_i^k = \upsilon, \tag{21}$$

$$c_i^k = x_i, \ \left[\sigma_{i1}^k, \sigma_{i2}^k\right] = \left[\sigma_i^k - \varsigma, \sigma_i^k + \varsigma\right], \tag{22}$$

From the 2nd rule onwards, it is checked if the newly arrived data (x_{i2} in Fig. 2b2, c2) can contribute to the generation of a new rule in the model. This is done by evaluating the degree of activation of each rule for the newly arrived data (Fig. 2b2, c2). If the firing strength does not exceed a predefined threshold, the existing rules are deemed insufficient as they cover the evidence of no coincidences of triggering of all input dimensions above the acceptable range. So, a new cluster is generated using recently arrived data (Fig. 2b3, c3). For this purpose, at every data arrival from the 2nd data onwards, IT2-GSETSK is partially run up to Layer III and the firing strength of each rule (Fig. 2b2, c2) is computed as in Eq. (10). The center of the firing strength interval of each rule k (r_c^k) is computed using Eq. (23) to obtain a unique number. Then the highest activated rule (ρ) is computed as in Eq. (24). Decision to create a new rule is taken if

$r_c^{\rho} \leq \theta$ where $\theta \in (0,1)$ is a firing threshold. The higher the firing threshold is set, the more rules are created. However, the interpretability is compromised, as discussed in section 2.3. The firing threshold is set to be set as 0.6 in this study.

$$r_c^k = \frac{1}{2}(\underline{r}^k + \bar{r}^k), \tag{23}$$

$$\rho = \arg \max_{1 \leq k \leq K(t)} r_c^k, \tag{24}$$

For the creation of new rule, for each input dimension i, MSGC evaluates the original fuzzy sets (IT1 versions) at that dimension to find the closest fuzzy set to the data (ε_i) as in Eq. (25). Then, the center of the closest GMF (c_i^{ε}) (Fig. 2b2, c2) is used to compute the STD of the new rule which is half of the distance of the new GMF center (x_i) and the closest one. This is specified in Eq. (26) in where $\eta = 0.5$ is the overlap degree, which prevents overlapping of GMFs. Bigger overlap degree results into bigger width for the newly-created fuzzy set.

$$\varepsilon_i = arg \max_{1 \leq k \leq K(t)} \exp\left[-\frac{1}{2}\left(\frac{x_i - c_i^k}{\sigma_i^k}\right)^2\right], \tag{25}$$

$$\sigma_i^k = \eta |x_i - c_i^{\varepsilon}|, \tag{26}$$

Ultimately, the STD computed using Eq. (26) and the center computed using Eq. (19) are used to form the new GMF using Eq. (21) for IT2-UM or Eq. (22) for IT2-USTD (Fig. 2b3, c3). After creation of the new rule, its similarity to the closest GMF is computed to consider if the new GMF should be retained as it is, discarded, or merged with the closest GMF. This ultimately ensures an up-to-date and compact rule base. The fuzzy subset measure [28] was used to measure the similarity (S) of two fuzzy sets A and B as in Eq. (27).

$$S = |A \cap B| / |A \cup B|, \tag{27}$$

where $\cap$ and $\cup$ are intersection and union. Based on the degree of the similarity the following actions will be taken:

- If $S > ThresA$, $ThresA = 0.8$ [29] then the new rule is discarded as the existing rule is representative
- If $S < ThresB$, $ThresB = 0.7$ [29] then the new rule is created
- If $ThresA > S > ThresB$ (Fig. 2b3, c3) then a new rule is created by merging the two rules

In the case of merging (Fig. 2b4, c4), Eq. (28) and Eq. (29) are used to create the new GMF.

$$C_{new} = (C_1 + C_2 + (\sigma_1 - \sigma_2)\sqrt{\pi})/2 , \tag{28}$$

$$\sigma_{new} = (C_1 - C_2 + (\sigma_1 + \sigma_2)\sqrt{\pi})/(2\sqrt{\pi}) , \tag{29}$$

where C_1 and C_2 are the centers and σ_1 and σ_2 are the STDs of the two GMFs. It is worth mentioning that the centers are average centers in case of using IT2-UM. Also, STDs are average STDs in the case of IT2-USTD. The merged GMF formed using Eq. (28-29) is IT1. It is converted to IT2 using Eq. (21) or Eq. (22) as required.

2.2.2. Unlearning Algorithm

The unlearning mechanism in IT2-GSETSK enables the model to maintain relevance and compactness of the rule base, which translates to higher accuracy and better interpretability, respectively. Unlearning is based on Hebbian learning mechanism [30] in which the synaptic connections are strengthened when pre-synaptic and post-synaptic activations occur simultaneously. For each rule in IT2-GSETSK, forward and backward firing strengths play the roles of pre-synaptic and post-synaptic activations, respectively. If both the forward and backward firing strength are highly activated at the same time, the rule is retained. The rule is pruned if the coincident activation of forward and backward firing strength is lower than a certain threshold. At each time t, the error of each rule k ($e_k(t)$) is calculated as in Eq. (30)

$$e_k(t) = |d(t) - f_k(X(t))|, \quad k = 1, \ldots, K(t), \tag{30}$$

where $d(t)$ is the desired output at time t, and $f_k(X(t))$ is the average of the linear equations defined in Eq. (14). A GMF with zero mean and STD as defined in Eq. (31) can measure the closeness of the output to the desired values. As illustrated in [27], such a GMF can be approximated by an isosceles triangle with unity height and bottom of $2\sigma_{back}(t)\sqrt{\pi}$.

$$\sigma_{back}(t) = \sum_{k=1}^{K(t)} e_k(t)/(K(t)\sqrt{\pi}), \tag{31}$$

The backward firing strength of k(th) rule, ($r_k^{back}(t)$) given in Eq. (32), shows the prediction accuracy of the rule. While the values close to one indicate high accuracy, values close to zero indicate low accuracy.

$$r_k^{back}(t) = \exp((-e_k(t)^2\ )/(2\sigma_{back}(t)^2\ )), \tag{32}$$

Then this backward firing strength $r_k^{back}(t)$ as well as the forward firing strength $r_k(x(t))$ is used for computation of the fuzzy rule potential $P_k(t)$ applicable for evaluation of the rule pruning in Eq. (33). The rule is pruned if $P_k(t)$ < *ThresP*. The greater value of *ThresP* cause more rules to be pruned. *ThresP* = 0.5 [26] was used in this paper.

$$P_k(t) = \gamma P_k(t-1) + r_k(x(t)) \times r_k^{back}(t), \quad k = 1, \dots, K(t), \tag{33}$$

where $P_k(t-1)$ is the previous fuzzy rule potential and $\gamma = [0.97, 0.99]$ [31] is the forgetting factor to let the rules gradually degrade. The higher values of this parameter lead to slower forgetting. $\gamma = 0.97$ [26] was used for all the experiments in this study.

2.3. Interpretability

Defining interpretability of NFMs is a controversial subject and still an open problem. This is because NFM are subjective and get affected by lots of factors such as number of rules, number of features, and shape of membership functions (MFs). The research conducted in [32] proposes a framework classifying different measures of interpretability based on the review of the other researches. Although this framework does not define which criteria are good, it provides a global vision to the researchers to come to their own conclusion about the interpretability of their model. Reducing the complexity and preserving the semantic associated with MFs are the key axes of interpretability. So, in this framework (Table 1), complexity-based interpretability and semantic-based interpretability are the two key measures which are considered both in the rule-base and fuzzy partition level resulting in four quadrants.

Table 1: Framework for analyzing interpretability(adapted from [32])

<table>
<tr><td rowspan="4">“Complexity-based interpretability”</td><td rowspan="2">“rule base level”</td><td rowspan="2">“Q1”</td><td>“number of rules”
(must be as small as possible while preserving the performance)</td></tr>
<tr><td>“number of conditions”
(number of distinct conditions in antecedent must not exceed 7±2)</td></tr>
<tr><td rowspan="2">“fuzzy partition level”</td><td rowspan="2">“Q2”</td><td>“number of features”
(to reduce dimensionality in high dimensional problems)</td></tr>
<tr><td>“number of MFs/labels (granularity)”
(should not exceed 7±2)</td></tr>
<tr><td rowspan="5">“Semantic-based Interpretability”</td><td rowspan="2">“rule base level”</td><td rowspan="2">“Q3”</td><td>“consistency of rules”
(absence of contradictory rules in RB)</td></tr>
<tr><td>“Number of rules fired at the same time”
(minimizing number of rules activated by a given input)</td></tr>
<tr><td rowspan="3">“fuzzy partition level”</td><td rowspan="3">“Q4”</td><td>“completeness or coverage”
(all universe of discourse needs to be covered at least by the MFs)</td></tr>
<tr><td>“normality”
(using normal fuzzy sets)</td></tr>
<tr><td>“distinguishability”
(clear semantic meaning and distinguishable from other MFs)</td></tr>
</table>

For each of these quadrants in Table 1, the main measures (and the favorable in parentheses) have been included. Following IT2-GSETSK status is cleared for each quadrant in Table 1:

- For limiting the number of rules and MFs (as a part of Q1 and Q2 in Table 1), there are the following mechanisms or constraints in-operation in IT2-GSETSK: 1) the firing threshold (θ, Fig. 2b2) limits the MF and hence rule generation; 2) similarity measure and merging mechanism (Fig. 2b3,b4,c3,c4) reduces the number of MFs and hence the number of rules; 3) rule pruning reduces the number of rules and MFs.
- Regarding the number of conditions (Q1) or features (Q2), there is no automatic way of handling it in IT2-GSETSK as dynamic feature selection is not present. Instead the number of features and conditions are limited by using expert-knowledge or static feature selection methods in advance.
- The firing threshold (θ) guarantees consistency of rules (Q3) with preventing the rule generation with the same premises. Also, the degree of overlap (η) dictates the rules to be generated in a way which guarantees less rules to be fired by a given input.
- As the mechanism of IT2-GSETSK creates new MFs whenever the existing MFs are not covering the newly-arrived data, the completeness of coverage (Q4) is ensured. Also, the distinguishability of MFs (Q4) is guaranteed by using the degree of overlap (η) between two

adjacent MFs. Finally, it is worth to mention that normality (Q4) is met in GSETSK by using all normal GMFs.

The interpretability of IT2-GSETSK will be discussed on different datasets in Section 4.

3. EXPERIMENTAL RESULTS

IT2-GSETSK was tested on 7 different data sets, namely the three well-known Nakanishi data sets, a stock market data set, an environmental data set and two medical data sets. These diverse data sets validate the utility of IT2-GSETSK to diverse practically important applications.

All parameters of IT2-GSETSK are common for different data sets as noted in detail in Section 2.2 (θ = 0.6, $\eta = 0.5$, $ThresA = 0.8$, $ThresB = 0.7$, $ThresP$ = 0.5, λ = 0.97), except the initial variance (υ) which controls the coverage of the first cluster. It is worth mentioning that assignment of large value to the initial variance (υ) is not critical because the MSGC clustering mechanism optimizes the size of clusters from the second incoming sample onwards. The recommended value of υ is roughly equal to one-fourth of the estimated range of the most relevant input variable range determined heuristically. The only other dataset-dependent parameter is the level of uncertainty assigned to IT2-GSETSK model. It may be assigned a value equal to the location or the coverage of the cluster, whichever is active. Since there is no prior knowledge of the type and level of uncertainty, all the datasets are tested under two different scenarios, namely uncertain mean (UM) and uncertain STD (USTD). For each scenario, the level of uncertainty is varied between 0.05-0.95 and the results are evaluated. This provides an insight about the choice of the type and level of uncertainty. For all of the data sets, IT2-GSETSK is executed fully online.

Performance of modeling is evaluated using criteria such as coefficient of efficiency (CE), coefficient of determination (R^2), root mean square error (RMSE), and mean absolute error (MAE) and interpretability. Additionally, statistical measures such as quantiles of absolute error and p-values are included for large datasets, i.e. excluding Nakanishi datasets.

In all the data sets, IT2-GSETSK is compared with GSETSK and the available benchmark results of other methods. The availability of NeuCom Student v0.919 software [33] facilitated benchmarking of the datasets for which benchmarks are unavailable in the literature. DENFIS [34] and EFuNN [35] are NFMs available in this package, representative of a TSK [10] and Mamdani [36] fuzzy system respectively. Since these models can learn continuously during the model's lifetime, they are good candidates for benchmarking IT2-GSETSK. However, as they are not fully online models and needs prior knowledge of data, they cannot be used for modeling the whole data. Instead, a part of data (as detailed for each dataset in Sections 3.1-3.5) is used in pre-training of these models before incremental learning can occur. We note that the best performance of DENFIS was achieved by assigning the value of distance threshold parameter, responsible for determining the cluster coverage, equal to 0.3. Also, the best performance of EFuNN was achieved by activation of pruning module of EuFNN. Details of the experiments and results on the seven datasets are presented in the following subsections.

3.1. The Three Nakanishi Datasets

Nakanishi datasets [37] include 3 datasets corresponding to a non-linear system, human operation of a chemical plant, and daily price of a stock in a stock market, respectively. These datasets have 50, 70 and 100 data samples respectively. Thus, they constitute examples of small datasets. In each of these, the first half of the data is used for model training and the remaining half for benchmarking in [38]. Nakanishi [37] compared different fuzzy reasoning methods including Sugeno P&P-G, Sugeno-P, Sugeno P-G, Mamdani, Turksen IVCRI (Table 2 represents the computations from [35]). While applying IT2-GSETSK and GSETSK on Nakanishi datasets, we used a fully online learning scheme, in comparison to the use of half of the data samples for pre-training as reported in [38]. However, for a comparison favorable to the methods that use pre-training, we use the same second half of the data used for benchmarking in [38] for reporting results on IT2-GSETSK and GSETSK. The benchmarking results for ANFIS [39], DENFIS, EFuNN, POP-CRI, RSPOP-CRI [38] are available in [38] and are recomputed and presented here for comparison. The details of experiments on Nakanishi dataset are discussed next.

3.1.1. A Non-linear System

This data set describes a nonlinear system behaving as in Eq. (34).

$$y = (1 + x_1^{-2} + x_2^{-1.5})^2, \quad (1 \leq x_1, x_2 \leq 5), \tag{34}$$

Sensitivity analysis of the model under the UM and USTD scenarios indicated that the best performance was achieved in USTD scenario with the level of uncertainty in the range [0.05, 0.5]. Table 2 shows the IT2-GSETSK results for USTD = 0.5. It is evident that IT2-GSETSK outperforms the other models by achieving R^2 and RMSE values of 0.92 and 0.34, respectively.

The 2nd best performance is observed for GSETSK with R^2 and RMSE values of 0.91 and 0.38 respectively. These results were achieved by a model with 6 rules. Notably, the performances of IT2-GSETSK and GSETSK results are far superior than the other models, for example, POP-CRI achieves R^2 and RMSE as 0.77 and 0.52 respectively using 192 rules.

3.1.2. Human Operation of a Chemical Plant

The 2nd Nakanishi data set is the human operation of a chemical plant. The output-input model for this dataset is stated in [37] as $y = f(x_1, x_3)$ where y is the set point for monomer, x_1 is the monomer concentration, and x_3 is the monomer flow rate.

The sensitivity analysis of IT2-GSETSK revealed that the best performance is achieved under the USTD scenario with the level of uncertainty in the range [0.5, 0.95]. Table 2 shows the results of IT2-GSETSK with USTD = 0.5. IT2-GSETSK outperforms the other methods, achieving the best performance values of R^2 and RMSE at 0.998 and 99, respectively. In terms of RMSE, IT2-GSETSK results in RMSE values of 41 and 130 units lesser than GSETSK and DENFIS respectively. It is worth mentioning that the reported performance for both IT2-GSETSK and GSETSK is obtained using only 3 rules.

Table 2: Modeling results of Nakanishi data sets. IT2-GSETSK uses USTD = 0.5.

	Nonlinear system		Chemical Plant		Stock market	
	R^2	RMSE	R^2	RMSE	R^2	RMSE
IT2-GSETSK	**0.92**	**0.34**	**0.998**	**99**	0.71	6.5
GSETSK	0.91	0.38	0.996	140	0.72	6.5
ANFIS	0.73	0.53	0.61	1723	0.77	6.2
DENFIS	0.65	0.64	0.99	229	0.66	8.4
EFuNN	0.52	0.75	0.89	851	0.57	8.5
POP-CRI	0.77	0.52	0.89	750	0.54	8.7
RSPOP-CRI	0.73	0.62	0.97	461	**0.85**	**5.0**
Sugeno P&P-G	0.69	0.59	0.96	537	0.50	9.7
Sugeno-P	0.73	0.58	0.88	787	0.78	5.9
Sugeno P-G	0.71	0.68	0.98	1389	0.49	13.0
Mamdani	0.69	0.66	0.88	797	0.75	6.4
Turksen IVCRI	0.37	0.84	0.986	508	0.48	9.5

3.1.3. Daily Price of a Stock in a Stock Market

The 3rd Nakanishi dataset is the daily stock market data of a particular stock. The output-input model used in [37] is $y = f(x_4, x_5, x_8)$ where y is the predicted stock price, x_4 is the present separation ratio with respect to moving average over a middle period, x_5 is the present change of moving over a short period, and x_8 is the past separation ratio with respect to the moving average over a short period.

The sensitivity analysis results of IT2-GSETSK indicate the best performance under the USTD scenario with the level of uncertainty lying in the range [0.4, 0.7]. The result of IT2-GSETSK with USTD = 0.5 presented in Table 2 shows that R^2 and RMSE are equal to 0.71 and 6.5, respectively. For this particular dataset, the performance of IT2-GSETSK is poorer than RSPOP-CRI, Sugeno-P, ANFIS, and Mamdani models. It is worth mentioning again that GSETSK/IT2-GSETSK is run in fully-online mode whereas the other models used the 1st half the data for pre-training. Pre-training the data gives advantage to the other models because of the reason we describe next. Nakanishi data set 3 has both inputs and outputs take values on either side of zero while the other 2 sets take positive values only. This property of dataset 3 upsets the parameter update of IT2-GSETSK.

3.2. Sydney Stock Market

A large dataset of Sydney stock market consisting of 8540 data samples was used in this analysis. This pertains the daily data of stock prices in 33 years during the working days from 3rd Jan 1985 to 14th Nov 2018 [40] (Fig. 3). At a given time instance, the price in the next time interval (y_1) is predicted using the current price (y_0), and the prices one, two, and three intervals ago (y_{-1}, y_{-2}, y_{-3}respectively), modelled as $y_1 = f(y_0, y_{-1}, y_{-2}, y_{-3})$.

The sensitivity analysis of IT2-GSETSK shows no significant difference in the results under different scenarios (UM and USTD) or with different levels of uncertainty. For benchmarking the results against EFuNN and DENFIS, the data of the first 5 years was used for pre-training and prediction was tested over the remaining years. Table 3 shows the results of the prediction over the last 28 years. As before, IT2-GSETSK and GSETSK were executed as fully online. There is no advantage of IT2-GSETSK over GSETSK for this dataset, as observed in Table 3. This may be because of more certainty of the inputs as the input variables are accurate historical data uncorrupted by noise. It is worth to mention, although DENFIS has a good prediction also, but this is as a cost of more rules up to 46 rules at the end of modeling. Also the fact that DENFIS requires pre-training should not be neglected. The excellent prediction demonstrated by IT2-GSETSK can also be witnessed qualitatively in Fig. 3.

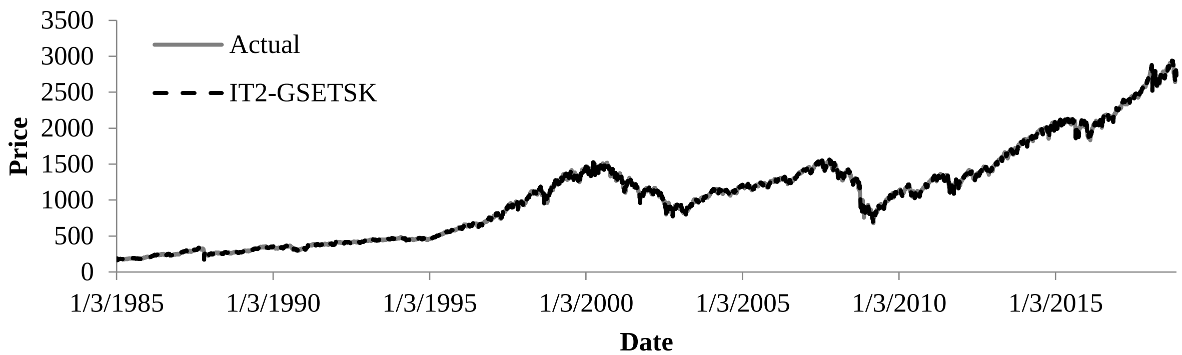

Fig. 3. Sydney stock market prediction between 1.Jan.1985 to 14.Nov.2018

Table 3: Modeling results of Sydney stock market over the last 28 years (7276 samples). [a]Quantile of absolute error. *If pairwise t-test rejects a null hypothesis with 5% significance

	IT2-GSETSK	GSETSK	DENFIS	EFuNN
CE	**1.00**	**1.00**	**1.00**	0.87
R^2	**1.00**	**1.00**	**1.00**	0.89
RMSE	13.3	**12.3**	14.7	217.1
MAE	8.7	**8.2**	9.4	82.5
Q[a]0.5	5.22	**5.20**	5.69	28.22
Q[a]0.9	**20.64**	20.77	22.21	150.42
P-value*	**0.07**	0.10	-	-
Rules	5	**2**	46	5

3.3. Patient Ventilator

Having a solution which can learn a prediction model using a small dataset is vital when dealing with medical applications. Proper control of ventilator in the life support systems of a patient is critical to maintain suitable oxygen supply to the patient. The requirement of each patient over time is unique to the patient's condition and cannot be pre-learnt, therefore imposing a significant challenge in automation of ventilator control systems. The online learning capability of IT2-GSETSK empowers it to perform the prediction from the second sample onward (see example data in Fig. 4). The hourly ventilation data (including 391 samples) of a patient is analyzed in this section. The fraction of oxygen to be supplied in the next hour ($FiO_2(t+1)$) is a function of the fraction of oxygen supplied in the current hour ($FiO_2(t)$) and the positive end expiratory pressure (PEEP) in the current hour (PEEP(t)) as $FiO_2(t+1) = f(FiO_2(t), PEEP(t))$.

The sensitivity analysis of IT2-GSETSK to different type and level of uncertainty shows no changes in the results. For benchmarking, the data of the first 30 hours was used for pre-training DENFIS and EFuNN models and continuous test and train was done for the remaining data. Table 4 shows the modeling results of FiO_2 predicted between hours 30-391. It is noted that all models except EFuNN perform well on this problem with IT2-GSETSK performing slightly better than the others by achieving lower RMSE as 1.20. However, beyond the similar performances of most approaches, IT2-GSETSK provides the ability to predict second sample onwards, a benefit that is critical in such applications.

Table 4: Modeling results of patient FiO_2 between hours 31 to 391 of the dataset (360 samples). [a] **Quantile of absolute error. *If pairwise t-test rejects a null hypothesis with 5% significance**

	IT2-GSETSK	GSETSK	DENFIS	EFuNN
CE	**0.93**	**0.93**	**0.93**	0.73
R^2	**0.93**	**0.93**	**0.93**	0.75
RMSE	**1.20**	1.23	1.24	2.35
MAE	0.54	0.54	**0.47**	1.14
Q[a]0.5	0.23	0.18	**0.13**	0.39
Q[a]0.9	1.00	1.02	**0.88**	4.25
P-value*	0.46	**0.41**	0.45	
Rules	8	**6**	9	8

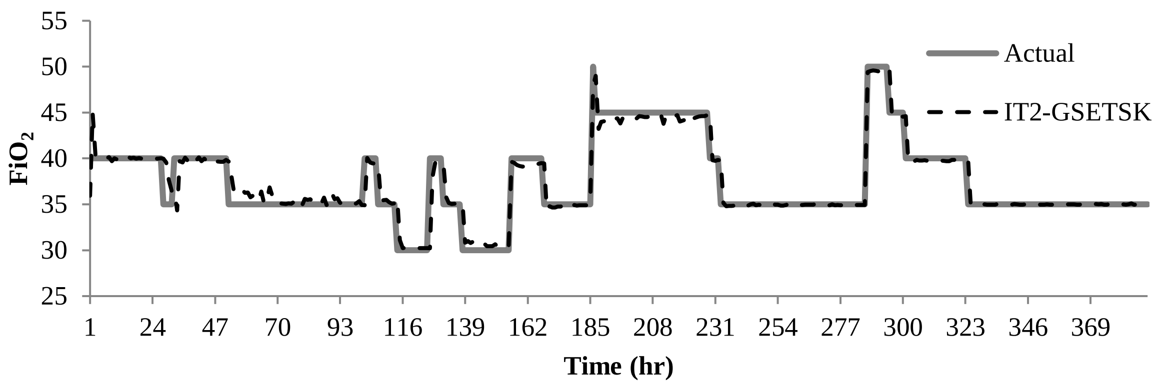

Fig. 4. FiO_2 modeled by IT2-GSETSK

3.4. Pulse Oximeter

The amount of oxygen carried in human body (SpO_2) can be predicted using the pulse rate data sensed from the fingertip. In this section, the current SpO_2 ($SpO_2(t)$) is modeled as a function of last observed SpO_2 ($SpO_2(t-1)$) and current sensed pulse rate ($P(t)$) as $SpO_2(t) = f(SpO_2(t-1), P(t))$.

The dataset includes 1811 samples. Fig. 5 shows the predicted SpO_2 results of IT2-GSETSK and demonstrates good agreement between the modeled and actual time series. It is worth mentioning that the type and level of uncertainty used in IT2-GSETSK results in no difference on the results. The first 200 data samples are used for pre-training DENFIS and EFuNN. The results are given in Table 5. It is evident that IT2-GSETSK outperforms the other models by achieving R^2 and RMSE values of 0.86 and 0.2, respectively. Interestingly, IT2-GSETSK uses a rule base of only 2 rules for modeling. The result of GSETSK is similar to IT2-GSETSK.

Table 5: Modeling results of SpO_2 (1610 samples). [a] Quantile of absolute error. *If pairwise t-test rejects a null hypothesis with 5% significance.

	IT2-GSETSK	GSETSK	DENFIS	EFuNN
CE	**0.86**	**0.86**	0.55	0.47
R^2	**0.86**	**0.86**	0.55	0.57
RMSE	**0.20**	**0.20**	0.36	0.39
MAE	**0.07**	**0.07**	0.23	0.18
Q^a0.5	**0.02**	**0.02**	0.11	0.04
Q^a0.9	**0.08**	**0.08**	0.62	1.00
P-value*	**0.36**	0.37	0.42	
Rules	**2**	**2**	3	16

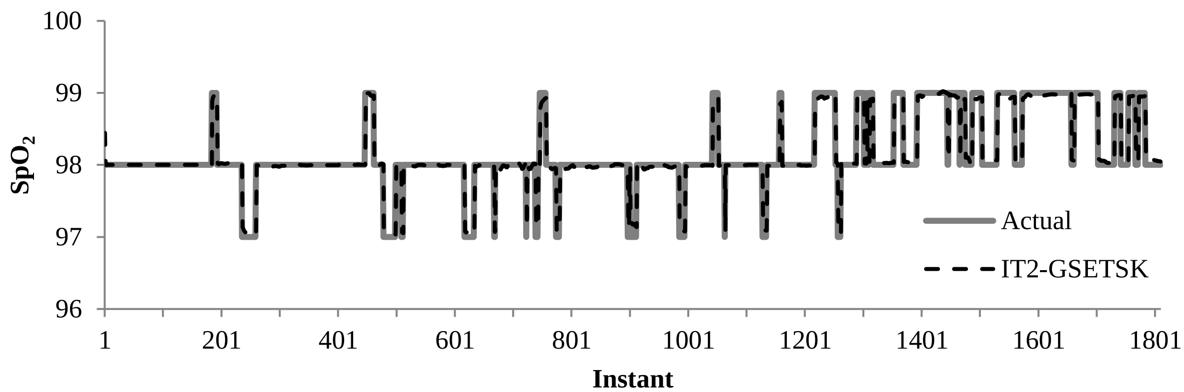

Fig. 5. SpO_2 modeled by IT2-GSETSK

3.5. Flood Forecasting

Flood forecasting is an important environmental application, which may help in disaster preparedness and management when used as an early warning system. Flood forecasting is done by modeling river discharge as a continuous process. In a recent research [26], GSETSK was applied for rainfall-runoff forecasting in a catchment in Sweden with limited availability of data. The river discharge in this region is a complex function of rainfall and temperature as some of the flow comes by snow melting. Based on the expert knowledge, 1-day-ahead river discharge (Q_1) is a function of current river discharge (Q_0), current precipitation (P_0) and current temperature (T_0) as $Q_1 = f(Q_0, P_0, T_0)$.

Two years of daily data including 731 samples were used for this model. The sensitivity analysis of IT2-GSETSK indicates that the best performance of the model is achieved for UM with level of uncertainty in the range [0.4, 0.7] and for USTD with level of uncertainty in the range [0.05, 0.5]. Table 6 and Fig. 6 show the results of IT2-GSETSK under USTD scenario with level of uncertainty set as 0.5. As DENFIS and EFuNN needs pre-training dataset, half of the data was used for this pre-training and the remaining half was used for benchmarking while being used for continuous testing and training. As obvious from Table 6, IT2-GSETSK provides an improvement of 2% in R^2 as compared to GSETSK. Also, it outperforms DENFIS, EFuNN and HBV [41]. Notably, HBV is the flood forecasting model currently being used by Swedish Meteorological and Hydrological Institute [42].

Table 6: Modeling results for river discharge modeling (365 samples). [a]Quantile of absolute error.

	IT2-GSETSK	GSETSK	DENFIS	EFuNN	HBV
CE	**0.88**	0.87	0.86	0.68	0.82
R^2	**0.89**	0.87	0.87	0.69	0.83
RMSE	**1.83**	1.89	1.98	2.92	2.21
MAE	**0.76**	0.78	0.80	1.30	1.36
Q[a]0.5	0.30	0.30	**0.24**	0.50	0.84
Q[a]0.9	**1.90**	**1.90**	1.99	3.20	3.09
Rules	6	**4**	8	36	-

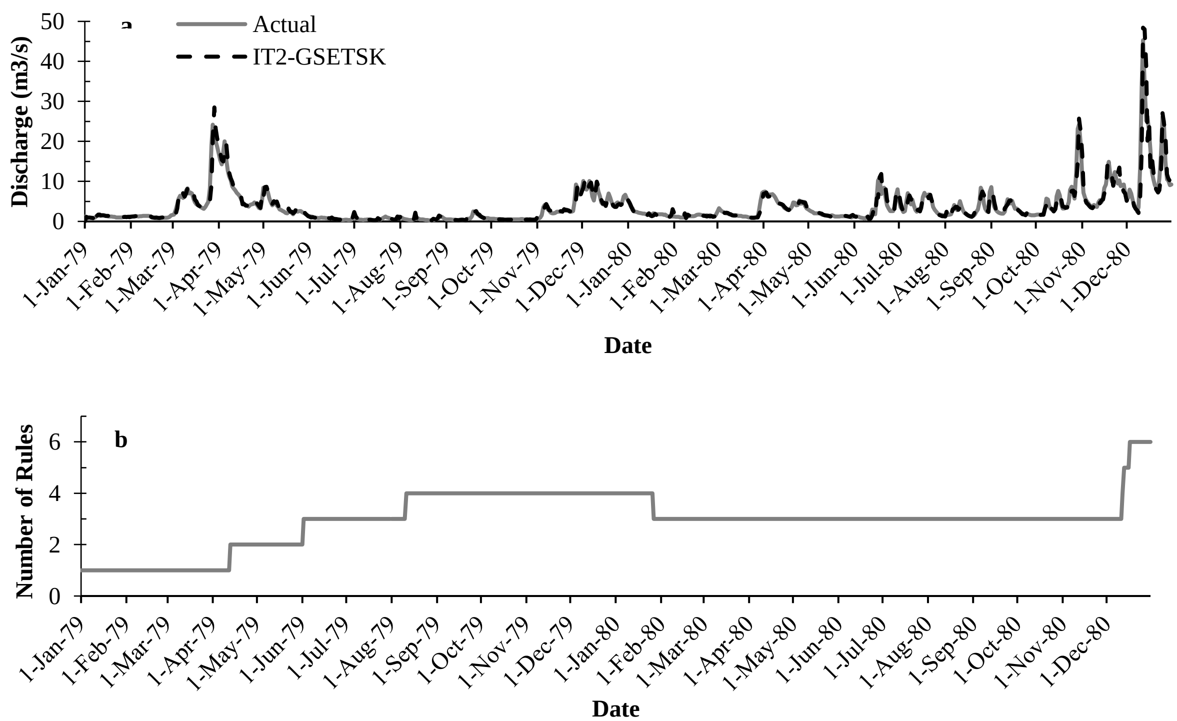

Fig. 6. River discharge forecasting (a) modeling result (b) rule base dynamics

4. DISCUSSION

Evaluation of the performance of IT2-GSETSK in different applications (Section 3) shows that IT2-GSETSK performs similar to, or better than the other methods (except for Nakanishi stock market data). This is in the case that some of these methods requiring pre-training while others are completely online. In fact, the fully-online mechanism of IT2-GSETSK and GSETSK, made it possible to work with data streams in an interleaving manner without pre-training data. The online learning and unlearning mechanisms of these two algorithms make them powerful to cope with any non-stationarity and time-variance of the problem. So, data challenges such as drift and shift will be handled. At the same time the learning and unlearning mechanisms ensures a model with few numbers of rules and membership functions which are easily interpretable. For IT2-GSETSK, additional feature of IT2 fuzzy membership copes with any uncertainties with the data.

Based on the experiments done on this research, using IT2-GSETSK with USTD = 0.5 is recommended for the best performance over noisy data.

Comparing the results of IT2-GSETSK to Type-I version (GSETSK), the best performances of IT2-GSETSK are for the chemical plant, non-linear system and flood data sets. Among them for the "human operation of a chemical plant" dataset, IT2-GSETSK had about % 31 improvement (in terms of RMSE) over GSETSK. This significant improvement possibly relates to more uncertainties with the data (possibly due to human operation) which IT2-GSETSK was able to handle it. For other SpO_2, FiO_2, and Sydney market price there is no advantages of IT2-GSETSK over GSETSK. This may be due to the nature of the problems in which some of them are more dynamic (relying on the inputs which are the outputs happened in the past). In this case, noise models are the same for the measurements of same quantities. On the other hand, for non-dynamic or less-dynamic problems, the noise models of different quantities may be different, resulting in predictions being afflicted by a variety of noise rather than the noise that affects the measurements. In summary, IT2-GSETSK is advantageous for handling noisy data of non-dynamic problems. However, when dealing with less noisy data representing and/or in dynamic problems, there is no significant outperformance of IT2-GSETSK over GSETSK. For such cases the usage of the simpler model, GSETSK is preferable.

As discussed in Section 2.3, IT2-GSETSK inherently meet some of the interpretability measures mentioned in Table 1. This is specifically true for semantic-based interpretability measures, namely consistency of rules, fewer number of rules fired at the same time, completeness of coverage, normality and distinguishability. Regarding complexity-based measures, for different experimental data, IT2-GSETSK meets these interpretability criteria (Table 7) also. In term of number of features or number of conditions, IT2-GSETSK results in a value less than 5 in all the experimental datasets. Also, in terms of number of rules, maximum number of 8 rules was achieved during the model's lifetime among all data set, which implicitly shows that the number of MFs are less than 8.

Table 7: Interpretability of IT2-GSETSK for different experimental datasets

Dataset	No. of features/ No. of conditions	No. of rules
Non-linear System	2	6
Human Operation of a Chemical Plant	2	3
Daily Price of a Stock in a Stock Market	3	3
Sydney Stock Market	4	5
Patient Ventilator	2	8
Pulse Oximeter	2	2
Flood Forecasting	3	6

The interpretability of IT2-GSETSK can be understood through the analysis of the rule base dynamics of the flood forecasting data. As discussed in Section 2, the specific clustering method, MSCG, in IT2-GSETSK has an important effect on the interpretability of the model. MSGC generates rule base of non-overlapping rules which is compact and interpretable. Also, the rule pruning mechanism in IT2-GSETSK makes it relevant and up-to-date. Fig. 6(b) shows the rule base dynamics of rainfall-runoff data when predicting the river discharge. As evident, the number of rules changes over time but is limited to a maximum of 6. The rule base history of rainfall-runoff data can be classified to the following stages.

1) Rule base foundation: In this stage, which lasts for more than half of the dataset up to end Jan 1980, the rule base is building the appropriate rules which number up to four.
2) Rule base pruning: In this stage, happening at almost the end of Jan 1980, an irrelevant rule is found and pruned.
3) Rule base update: After Mid Dec 1980, 3 new rules are added to cope with the new data trends, such as the occurrence of the highest peak in the time series.

Finally, Fig.7 shows IT2 GMFs for each input space dimension (river discharge, precipitation and temperature) at the end of the model's lifetime. As evident, there are limited numbers of

GMFs (2) in each input dimension space. These GMFs can be easily assigned as human-interpretable tags, for example, less or more river discharge, light or heavy precipitation, and cold and warm temperature.

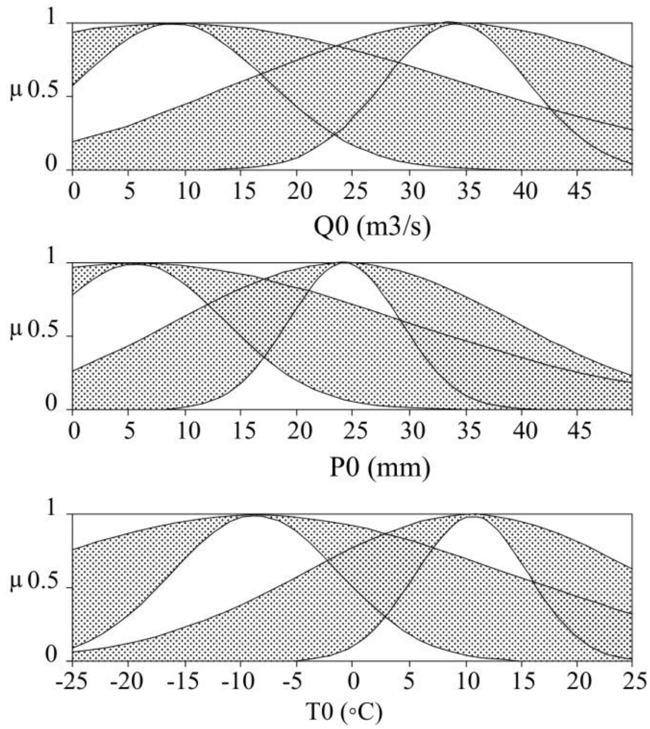

Fig. 7. IT2-GSETSK MFs for each input space at the end of model river discharge.

5. CONCLUSION

IT2-GSETSK was found to be advantageous for modeling noisy data of non-dynamic problem. For dynamic problems as well as problems with less noisy data, the simpler model, GSETSK is enough. Both two models were able to compete the benchmarks quantitatively and qualitatively (in terms of interpretability of neuro-fuzzy models). This competence is due to specific learning and unlearning ability of the models and the presence of IT2 membership functions (for IT2-GSETSK). For the next stage, the authors target to imbed an online feature selection method to the algorithm which alleviate the current need of separate feature selection. Also, online optimization of the clusters is another future work for this algorithm.

List of symbols

i	Input dimension index
k	Rule number Index
n	Number of Input dimensions
$K(t)$	Number of rules at time *t*
x_i	Input value at dimension *i*
R^k	*k(th)* rule
$\tilde{A}_i^k$	IT2 fuzzy set A (*k(th)* rule in the input dimension *i*)
$\underline{\mu}_{\tilde{A}_i^k}$ and $\bar{\mu}_{\tilde{A}_i^k}$	Degree of membership lower and upper bounds for fuzzy set A, respectively
c_i and σ_i	Center and standard deviation of fuzzy set
r^k	Interval firing strength
$\underline{r}^k$ and $\bar{r}^k$	Lower and upper degrees of activation at rule *k*
Ω^k	Interval normalized firing strengths
$\underline{\omega}^k$ and $\overline{\omega}^k$	Normalized firing strengths' lower and upper bounds
$F^k(X)$	Interval consequent node output
$\underline{f}^k(X)$, $\bar{f}^k(X)$	Consequent node linear equation's lower and upper bounds
$\tilde{b}_i^k$	interval consequent parameter that relates to input dimension *i* of *k(th)* rule
$\underline{q}^k$ and $\bar{q}^k$	Consequent node output's lower and upper bounds
y	Final Output
υ	Predefined value for the 1st cluster
α	Degree of uncertainty
r_c^k	Center of the firing strength interval of rule *k*
ρ	Rule which has the maximum firing strength
θ	Firing Threshold
ε	Largest matching degree
η	Degree of overlap between two GMF
S	Measure of similarity
$ThresA$	Threshold for rule base management
$ThresB$	Threshold for rule base management
$d(t)$	Desired output
$e_k(t)$	Error of node K at time t
$r_k^{\mathrm{back}}(t)$	Backward firing strength of node K at time t
$r_k(\mathrm{x}(t))$	Forward firing strength of node K at time t
$P_k(t)$	Fuzzy rule potential
Γ	Forgetting factor

REFERENCES

[1] L.A. Zadeh, The concept of a linguistic variable and its application to approximate reasoning—I, Information Sciences, 8 (1975) 199-249.

[2] J.M. Mendel, Computing derivatives in interval type-2 fuzzy logic systems, IEEE Transactions on Fuzzy Systems, 12 (2004) 84-98.

[3] W.M. Lewis, Basis for the protection and management of tropical lakes, Lakes & Reservoirs: Research & Management, 5 (2000) 35-48.

[4] N.N. Karnik, J.M. Mendel, L. Qilian, Type-2 fuzzy logic systems, IEEE Transactions on Fuzzy Systems, 7 (1999) 643-658.

[5] N. Baklouti, A. Abraham, A.M. Alimi, A Beta basis function Interval Type-2 Fuzzy Neural Network for time series applications, Engineering Applications of Artificial Intelligence, 71 (2018) 259-274.

[6] M. Han, K. Zhong, T. Qiu, B. Han, Interval Type-2 Fuzzy Neural Networks for Chaotic Time Series Prediction: A Concise Overview, IEEE Transactions on Cybernetics, (2018) 1-12.

[7] N.K. Kasabov, Evolving connectionist systems for adaptive learning and knowledge discovery: Trends and directions, Knowledge-Based Systems, 80 (2015) 24-33.

[8] P.F. Angelov, D.P.; Kasabov, N., Evolving intelligent systems: methodology and applications (Wiley-IEEE Press, 2010).

[9] G. Ditzler, M. Roveri, C. Alippi, R. Polikar, Learning in Nonstationary Environments: A Survey, IEEE Computational Intelligence Magazine, 10 (2015) 12-25.

[10] T. Takagi, M. Sugeno, Fuzzy identification of systems and its applications to modeling and control, IEEE Transactions on Systems, Man, and Cybernetics, SMC-15 (1985) 116-132.

[11] C. Juang, Y. Tsao, A Type-2 Self-Organizing Neural Fuzzy System and Its FPGA Implementation, IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics), 38 (2008) 1537-1548.

[12] C. Juang, Y. Tsao, A Self-Evolving Interval Type-2 Fuzzy Neural Network With Online Structure and Parameter Learning, IEEE Transactions on Fuzzy Systems, 16 (2008) 1411-1424.

[13] C. Juang, R. Huang, Y. Lin, A Recurrent Self-Evolving Interval Type-2 Fuzzy Neural Network for Dynamic System Processing, IEEE Transactions on Fuzzy Systems, 17 (2009) 1092-1105.

[14] C. Juang, R. Huang, W. Cheng, An Interval Type-2 Fuzzy-Neural Network With Support-Vector Regression for Noisy Regression Problems, IEEE Transactions on Fuzzy Systems, 18 (2010) 686-699.

[15] Y. Lin, J. Chang, N.R. Pal, C. Lin, A Mutually Recurrent Interval Type-2 Neural Fuzzy System (MRIT2NFS) With Self-Evolving Structure and Parameters, IEEE Transactions on Fuzzy Systems, 21 (2013) 492-509.

[16] Y. Lin, J. Chang, C. Lin, A TSK-Type-Based Self-Evolving Compensatory Interval Type-2 Fuzzy Neural Network (TSCIT2FNN) and Its Applications, IEEE Transactions on Industrial Electronics, 61 (2014) 447-459.

[17] C. Juang, K. Juang, Reduced Interval Type-2 Neural Fuzzy System Using Weighted Bound-Set Boundary Operation for Computation Speedup and Chip Implementation, IEEE Transactions on Fuzzy Systems, 21 (2013) 477-491.

[18] A.K. Das, S. Sundaram, N. Sundararajan, A Self-Regulated Interval Type-2 Neuro-Fuzzy Inference System for Handling Nonstationarities in EEG Signals for BCI, IEEE Transactions on Fuzzy Systems, 24 (2016) 1565-1577.

[19] M. Pratama, J. Lu, E. Lughofer, G. Zhang, M.J. Er, An Incremental Learning of Concept Drifts Using Evolving Type-2 Recurrent Fuzzy Neural Networks, IEEE Transactions on Fuzzy Systems, 25 (2017) 1175-1192.

[20] C. Za'in, M. Pratama, E. Lughofer, S.G. Anavatti, Evolving type-2 web news mining, Applied Soft Computing, 54 (2017) 200-220.

[21] A.K. Das, K. Subramanian, S. Sundaram, An Evolving Interval Type-2 Neurofuzzy Inference System and Its Metacognitive Sequential Learning Algorithm, IEEE Transactions on Fuzzy Systems, 23 (2015) 2080-2093.

[22] M. Pratama, E. Lughofer, M.J. Er, S. Anavatti, C.-P. Lim, Data driven modelling based on Recurrent Interval-Valued Metacognitive Scaffolding Fuzzy Neural Network, Neurocomputing, 262 (2017) 4-27.

[23] M. Pratama, G. Zhang, M.J. Er, S. Anavatti, An Incremental Type-2 Meta-Cognitive Extreme Learning Machine, IEEE Transactions on Cybernetics, 47 (2017) 339-353.

[24] H.-G. Han, J.-M. Li, X.-L. Wu, J.-F. Qiao, Cooperative strategy for constructing interval type-2 fuzzy neural network, Neurocomputing, 365 (2019) 249-260.

[25] H.-G. Han, Z.-Y. Chen, H.-X. Liu, J.-F. Qiao, A self-organizing interval Type-2 fuzzy-neural-network for modeling nonlinear systems, Neurocomputing, 290 (2018) 196-207.

[26] M. Ashrafi, L.H.C. Chua, C. Quek, X. Qin, A fully-online Neuro-Fuzzy model for flow forecasting in basins with limited data, Journal of Hydrology, 545 (2017) 424-435.

[27] N.N. Nguyen, W.J. Zhou, C. Quek, GSETSK: a generic self-evolving TSK fuzzy neural network with a novel Hebbian-based rule reduction approach, Applied Soft Computing, 35 (2015) 29-42.

[28] J. Yen, R. Langari, Fuzzy logic : intelligence, control, and information (Upper Saddle River, N.J. : Prentice Hall, c1999., 1999).

[29] I.B. Turksen, Z. Zhong, An Approximate Analogical Reasoning Approach Based on Similarity Measures, Ieee T Syst Man Cyb, 18 (1988) 1049-1056.

[30] J.R. Whitlock, A.J. Heynen, M.G. Shuler, M.F. Bear, Learning induces long-term potentiation in the hippocampus, Science, 313 (2006) 1093-1097.

[31] W.L. Tung, C. Quek, eFSM-A Novel Online Neural-Fuzzy Semantic Memory Model, Ieee Transactions on Neural Networks, 21 (2010) 136-157.

[32] M.J. Gacto, R. Alcalá, F. Herrera, Interpretability of linguistic fuzzy rule-based systems: An overview of interpretability measures, Information Sciences, 181 (2011) 4340-4360.

[33] N. Kasabov, NeuCom - Knowledge Engineering and Discovery Research Institute - AUT, 2017).

[34] N.K. Kasabov, S. Qun, DENFIS: dynamic evolving neural-fuzzy inference system and its application for time-series prediction, IEEE Transactions on Fuzzy Systems, 10 (2002) 144-154.

[35] N. Kasabov, Evolving fuzzy neural networks for supervised/unsupervised online knowledge-based learning, IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics), 31 (2001) 902-918.

[36] E.H. Mamdani, S. Assilian, An experiment in linguistic synthesis with a fuzzy logic controller, International Journal of Man-Machine Studies, 7 (1975) 1-13.

[37] H. Nakanishi, I.B. Turksen, M. Sugeno, A review and comparison of six reasoning methods, Fuzzy Sets and Systems, 57 (1993) 257-294.

[38] A. Kai Keng, Q. Chai, RSPOP: Rough Set–Based Pseudo Outer-Product Fuzzy Rule Identification Algorithm, Neural Computation, 17 (2005) 205-243.

[39] J.S.R. Jang, ANFIS: adaptive-network-based fuzzy inference system, IEEE Transactions on Systems, Man, and Cybernetics, 23 (1993) 665-685.

[40] A.R. Iyer, D.K. Prasad, C.H. Quek, PIE-RSPOP: A brain-inspired pseudo-incremental ensemble rough set pseudo-outer product fuzzy neural network, Expert Systems with Applications, 95 (2018) 172-189.

[41] S. Bergstrom , A. Forsman, Development of a conceptual deterministic rainfall-runoff model, Hydrology Research, 4 (1973) 147-170.

[42] SMHI, http://www.smhi.se/Professionella-tjanster/Professionella-tjanster/Miljo-och-klimat/Vattenmiljo/referensguide-till-smhi-vattenwebb-1.22742, 2013).

Mohammad Ashrafi is a post doctoral research fellow with iTrust Centre for Research in Cyber Security, Singapore University of Technology and Design (SUTD). He received the B.Sc., M.Sc. and PhD degrees in geology, hydrogeology, and civil and environmental engineering from Isfahan University, Iran, Shahid Beheshti University, Iran, and Nanyang Technological University, Singapore in 2002, 2008 and 2017 respectively. His research interests are security of cyber physical systems, data analytics and model development, artificial intelligence (AI), and computational intelligence.

Dilip K. Prasad is with Department of Computer Science, UiT The Arctic University of Norway, Tromso. He received B.Tech. and Ph.D. degrees in computer science and engineering from the Indian Institute of Technology (ISM), Dhanbad, India, and Nanyang Technological University, Singapore, respectively. He has co-authored over 100 internationally peer-reviewed research articles.

Chai Quek is with the School of Computer Science and Engineering, Nanyang Technological University, Singapore. He received B.Sc. and Ph.D. degrees from Heriot-Watt University, Edinburgh, U.K. His research interests include neurocognitive informatics, biomedical engineering and computational finance. He has published over 250 international conference and journal papers. He has been invited as a Program Committee Member and reviewer for several conferences and journals, including