

Elsevier required licence: © <2018>. This manuscript version is made available under the CC-BY-NC-ND 4.0 license <http://creativecommons.org/licenses/by-nc-nd/4.0/>  
The definitive publisher version is available online at

[<https://www.sciencedirect.com/science/article/pii/S1877050918322117?via%3Dihub>]

INNS Conference on Big Data and Deep Learning 2018

# Wave Forecasting using Meta-cognitive Interval Type-2 Fuzzy Inference System

Nguyen Anh<sup>a</sup>, Mukesh Prasad<sup>b</sup>, Narasimalu Srikanth<sup>a</sup>, Suresh Sundaram<sup>a,\*</sup><sup>a</sup>Nanyang Technological University, Singapore<sup>b</sup>Centre for Artificial Intelligence, School of Software, FEIT, University of Technology Sydney, Australia

---

## Abstract

Renewable energy is fast becoming a mainstay in today's energy scenario. One of the important sources of renewable energy is the wave energy, in addition to wind, solar, tidal, etc. Wave prediction/forecasting is consequently essential in coastal and ocean engineering studies. However, it is difficult to predict wave parameters in long term and even in the short term due to its intermittent nature. This study aims to propose a solution to handle the issue using Interval type-2 fuzzy inference system, or IT2FIS. IT2FIS has been shown to be capable of handling uncertainty associated with the data. The proposed IT2FIS is a fuzzy neural network realizing Takagi-Sugeno-Kang inference mechanism employing meta-cognitive learning algorithm. The algorithm monitors knowledge in a sample to decide an appropriate learning strategy. Performance of the system is evaluated by studying significant wave heights obtained from buoys located in Singapore. The results compared with existing state-of-the art fuzzy inference system approaches clearly indicate the advantage of IT2FIS based wave prediction.

© 2018 The Authors. Published by Elsevier Ltd.

This is an open access article under the CC BY-NC-ND license (<https://creativecommons.org/licenses/by-nc-nd/4.0/>)

Selection and peer-review under responsibility of the INNS Conference on Big Data and Deep Learning 2018.

**Keywords:** wave prediction, interval type-2 fuzzy systems, meta-cognition, long-term forecast

---

## 1. Introduction

Renewable energy is becoming more acceptable as an alternative source of energy. They provide more environmental friendly and cheaper power but the electrical power generation is becoming more complex with inclusion of these sources. One of the main reasons for the complexity is the inability to accurately predict the strength of these sources at a given time. There are various natural as well as artificial causes. As a result, the forecast of the energy generated is very uncertain. This uncertainty leads to unpredictable or unrealistic generation, even leads to financial losses. Hence, realistic forecast of these sources is the need for increased and improved renewable energy usage. In this study, we attempt to forecast wave energy by working on an important characteristic of wave, namely significant wave height. Recently, artificial neural network has been used in predicting wave height [15, 9]. It has been

---

\* Corresponding author. Tel.: +65 6790 6185.

E-mail address: [ssundaram@ntu.edu.sg](mailto:ssundaram@ntu.edu.sg) (S. Suresh)

shown in the literature that artificial neural network could be developed with fuzzy inference system to enhance the learning ability. It is referred to as neuro-fuzzy inference system [11, 2]. Traditionally, neuro-fuzzy inference system employs type-1 fuzzy sets, which have certain membership functions and are unable to handle uncertainty. To solve the problem, Zadeh has proposed type-2 fuzzy sets and type-2 fuzzy logic system which employs these type-2 sets [10]. Type-2 fuzzy sets are able to deal with uncertainty but the computation is massive. As a result, interval type-2 fuzzy sets are proposed to minimize the computational effort in handling uncertainty in data [12]. Based on these Interval Type-2 fuzzy sets, different fuzzy inference systems, known as interval type-2 fuzzy inference system or IT2FIS, and their learning algorithms have been proposed in literature [16, 5, 8, 13, 7, 6]. Among all of the above mentioned IT2FIS, those employing meta-cognitive learning mechanism have shown an effective generalization capability. Meta-cognition is known as the ability of human brain to make decision on how to learn a specific knowledge by using suitable learning strategies. Various studies on meta-cognitive algorithm with neuro-fuzzy inference system in the literature have clearly shown their efficient performance [20, 7, 6, 8, 19, 4, 3, 18, 17].

The interval type-2 neuro fuzzy inference system proposed in this study is based on an Extended Kalman filtering and employs meta-cognitive learning for significant wave height forecasting problem. In this work, we call it McIT2FIS for wave height. The learning mechanism of McIT2FIS is formulated on a five-layer network realizing Takagi Sugeno Kang fuzzy inference. The input layer has  $n$  nodes, followed by  $K$  nodes of the membership layer. The firing layer calculates the firing strength of  $K$  rules. The interval reduction layer uses the two  $q$  factors referred in [1] to construct output in the output layer. Meta-cognition inspires the learning mechanism of McIT2FIS. As an input arrives, meta-cognitive component decides whether to delete the sample, learn the sample or reserve the sample based on its knowledge. These three decisions are crucial part of meta-cognitive leaning algorithm.

In section II, the architecture of the studied interval type-2 neuro fuzzy system is presented. Meta-cognitive mechanism and EKF-based algorithm are deployed in section III followed by performance evaluation on wave forecasting problem in section IV. Section V concludes the main features of the paper.

## 2. Meta-cognitive interval type-2 neuro fuzzy inference system

Figure 1 is the proposed meta-cognitive interval type-2 inference system for wave height forecasting. The structure consists of five layers. The aim is to estimate the relationship between the input and the output. Assume that the system consists of  $m$  input features and has grown  $K$  rules. The function of each layer with the  $t$ -th sample,  $x(t)$  is presented belows.

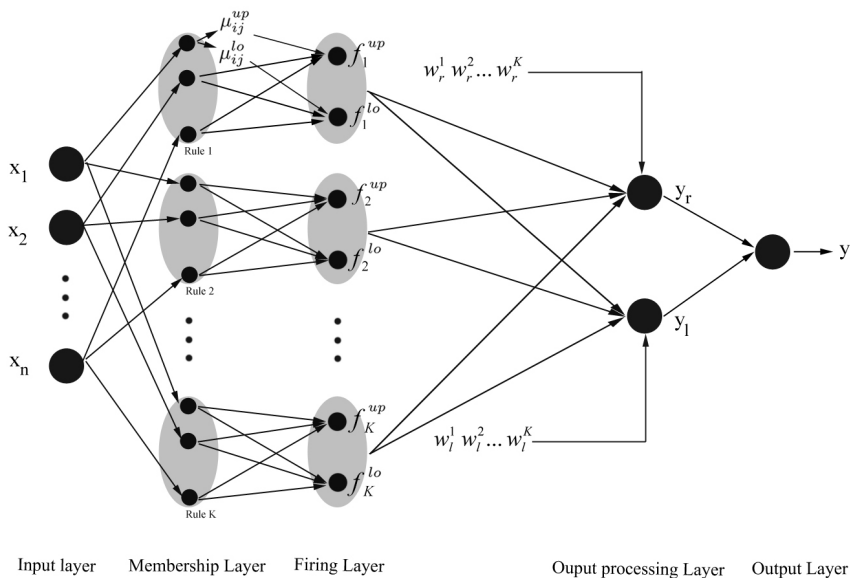


Fig. 1: Architecture of the interval type-2 neuro fuzzy system.

**Layer 1-** Input layer: This layer contains  $n$  nodes representing  $n$  number of input features. The input is passed directly to layer 2, Membership Function Layer. The output of  $j$ -th node is given:

$$u_j(t) = x_j(t); j = 1, 2, \dots, n. \quad (1)$$

**Layer 2-** Fuzzification layer: This layer calculates the upper and the lower membership strength of each  $j$ -th feature in every  $i$ -th rule. The formulas are given by:

$$\mu_{ij}^{up}(x_j) = \begin{cases} \phi(m_{j1}^i, \sigma_j^i, x_j) & x_j < m_{j1}^i \\ 1 & m_{j1}^i \leq x_j \leq m_{j2}^i \\ \phi(m_{j2}^i, \sigma_j^i, x_j) & x_j > m_{j2}^i \end{cases} \quad (2)$$

$$\mu_{ij}^{lo}(x_j) = \begin{cases} \phi(m_{j2}^i, \sigma_j^i, x_j) & x_j \leq \frac{m_{j1}^i + m_{j2}^i}{2} \\ \phi(m_{j1}^i, \sigma_j^i, x_j) & x_j > \frac{m_{j1}^i + m_{j2}^i}{2} \end{cases} \quad (3)$$

where,

$$\phi(m_j^i, \sigma_j^i, x_j) = \exp\left(-\frac{(x_j - m_j^i)^2}{2(\sigma_j^i)^2}\right) \quad (4)$$

where,  $m_{j1}, m_{j2}, \sigma$  are center left, center right and width of the interval type-2 Gaussian membership function accordingly.

**Layer 3-** Firing layer: The firing strength of each rule is calculated in this layer based on the following formulas:

$$f_i^{lo} = \prod_{j=1}^n \mu_{ij}^{lo}; f_i^{up} = \prod_{j=1}^n \mu_{ij}^{up} \quad (5)$$

**Layer 4-** Output processing layer: This layer has  $K$  nodes, each of which represents a rule in the network. Instead of using Karnick-Mendel iterative procedure to find lower and upper end points, this study uses  $q$  factors to increase the learning speed and minimize the computational complexity. The two end points are given as:

$$y_l = \frac{(1 - q_l) \sum_{i=1}^K f_i^{up} w_l^i + q_l \sum_{i=1}^K f_i^{lo} w_l^i}{\sum_{i=1}^K (f_i^{lo} + f_i^{up})} \quad (6)$$

$$y_r = \frac{(1 - q_r) \sum_{i=1}^K f_i^{lo} w_r^i + q_r \sum_{i=1}^K f_i^{up} w_r^i}{\sum_{i=1}^K (f_i^{lo} + f_i^{up})} \quad (7)$$

**Layer 5-** Output layer: The output of this layer is the sum of the two end points from layer 4:

$$\hat{y} = y_l + y_r \quad (8)$$

### 3. Meta-cognitive learning algorithm for IT2FIS

The objective is to estimate the function  $f[.]$  so that the predicted output:

$$\hat{y} = f[x(t), \theta] \quad (9)$$

approximates the actual output with parameter vector  $\theta$ .

In order to measure the novelty in the current sample, prediction error and spherical potential are considered. The error for  $t$ -th sample is given by:

$$e(t) = y(t) - \hat{y}(t) \quad (10)$$

Prediction error and spherical potential are respectively given by:

$$E(t) = \sqrt{e^2(t)} \quad (11)$$

$$\psi(t) = \sum_{i=1}^K \frac{F_i^{lo}(t) + F_i^{up}(t)}{2K} \quad (12)$$

### 3.1. Sample delete strategy

The sample will be deleted if it contains knowledge that is not novel in the system or its prediction error is smaller than the delete threshold,  $E_d$ . The criterion is given as:

$$E(t) < E_d \quad (13)$$

### 3.2. Sample learn strategy

The new rule is added into the system if these existing rules cannot cover new sample effectively. Thus, we will define the rule addition criterion to checks against the prediction error presented by the existing rules as well as the novelty of the sample

#### 3.2.1. The Rule Adding Criterion

$$E(t) > E_a \quad \text{AND} \quad \psi(t) < E_s \quad (14)$$

where,  $E_a$  is the adding threshold and  $E_s$  is the novelty threshold. The new rule  $K+1$  centers is initialized as:

$$[m_{j1}^{K+1}, m_{j1}^{K+1}] = [x_j - 0.1, x_j + 0.1] \quad (15)$$

The width is given as

$$[\sigma^{K+1}] = \min_j [\|x(t) - \mu_{j1}\|, \|x(t) - \mu_{j2}\|] \quad (16)$$

#### 3.2.2. The Rule Updating Criterion

The rule update criterion is as follows:

$$E_u < E(t) < E_a \quad \text{AND} \quad \psi(t) > E_s \quad (17)$$

The parameters are updated by an Extended Kalman filtering algorithm. EKF has been applied in various learning algorithm and has shown its computationally fast performance [7, 20]. The formula updating parameters for this problem are given:

$$\theta = \theta + e(t)G^T. \quad (18)$$

where,  $\theta = [m_1, m_2, \sigma, w, q]$  is the the parameter vector of the network,  $e(t)$  is the error, and  $G$  is the EKF gain matrix [7].  $G$  is given by:

$$G = PH[R + H^T PH]^{-1} \quad (19)$$

where,  $P$  is the error covariance matrix,  $R = r_0 I$  is the variance of measurement noise, and  $H$  is the gradient matrix.  $P$  is initialized as  $P = p_0 I$  and is updated as:

$$P = [I - GH^T]P + q_0 I \quad (20)$$

It should be noted that  $I$  is the identity matrix. The gradient of predicted output with respect to network parameters are described below.

The equations for updating  $w_l^i$  are as follows:

$$\frac{\partial y_l}{\partial w_l^i} = \frac{(1 - q_l)f_i^{lo} + q_l f_i^{up}}{\sum_{i=1}^K (f_i^{lo} + f_i^{up})} \quad (21)$$

Similarly, the updating equation for  $w_r^i$  is given:

$$\frac{\partial y_r}{\partial w_r^i} = \frac{(1 - q_r)f_i^{lo} + q_r f_i^{up}}{\sum_{i=1}^K (f_i^{lo} + f_i^{up})} \quad (22)$$

The derivations of the premise part are expressed by the following:

**For the left means:**

$$\frac{\partial y_l + \partial y_r}{\partial m_{j1}^i} = \left( \frac{\partial y_l}{\partial f_i^{up}} + \frac{\partial y_r}{\partial f_i^{up}} \right) \frac{\partial f_i^{up}}{\partial m_{j1}^i} + \left( \frac{\partial y_l}{\partial f_i^{lo}} + \frac{\partial y_r}{\partial f_i^{lo}} \right) \frac{\partial f_i^{lo}}{\partial m_{j1}^i} \quad (23)$$

**For the right means:**

$$\frac{\partial y_l + \partial y_r}{\partial m_{j2}^i} = \left( \frac{\partial y_l}{\partial f_i^{up}} + \frac{\partial y_r}{\partial f_i^{up}} \right) \frac{\partial f_i^{up}}{\partial m_{j2}^i} + \left( \frac{\partial y_l}{\partial f_i^{lo}} + \frac{\partial y_r}{\partial f_i^{lo}} \right) \frac{\partial f_i^{lo}}{\partial m_{j2}^i} \quad (24)$$

**For the width:**

$$\frac{\partial y_l + \partial y_r}{\partial \sigma_j^i} = \left( \frac{\partial y_l}{\partial f_i^{up}} + \frac{\partial y_r}{\partial f_i^{up}} \right) \frac{\partial f_i^{up}}{\partial \sigma_j^i} + \left( \frac{\partial y_l}{\partial f_i^{lo}} + \frac{\partial y_r}{\partial f_i^{lo}} \right) \frac{\partial f_i^{lo}}{\partial \sigma_j^i} \quad (25)$$

where,

$$\frac{\partial y_l}{\partial f_i^{up}} = \frac{(1 - q_l)w_l^i - y_l}{\sum_{i=1}^K (f_i^{lo} + f_i^{up})}, \quad \frac{\partial y_l}{\partial f_i^{lo}} = \frac{q_l w_l^i - y_l}{\sum_{i=1}^K (f_i^{lo} + f_i^{up})}; \quad (26)$$

$$\frac{\partial y_r}{\partial f_i^{up}} = \frac{q_r w_r^i - y_r}{\sum_{i=1}^K (f_i^{lo} + f_i^{up})}, \quad \frac{\partial y_r}{\partial f_i^{lo}} = \frac{(1 - q_r)w_r^i - y_r}{\sum_{i=1}^K (f_i^{lo} + f_i^{up})}; \quad (27)$$

$$\frac{\partial f_i^{up}}{\partial m_{j1}^i} = \frac{\partial f_i^{up}}{\partial \mu_{ij}^{up}} \frac{\partial \mu_{ij}^{up}}{\partial m_{j1}^i} = \begin{cases} f_i^{up} \frac{x_j - m_{j1}^i}{(\sigma_j^i)^2} & x_j \leq m_{j1}^i \\ 0 & \text{otherwise} \end{cases} \quad (28)$$

$$\frac{\partial f_i^{lo}}{\partial m_{j1}^i} = \frac{\partial f_i^{lo}}{\partial \mu_{ij}^{lo}} \frac{\partial \mu_{ij}^{lo}}{\partial m_{j1}^i} = \begin{cases} f_i^{lo} \frac{x_j - m_{j1}^i}{(\sigma_j^i)^2} & x_j > \frac{m_{j1}^i + m_{j2}^i}{2} \\ 0 & \text{otherwise} \end{cases} \quad (29)$$

$$\frac{\partial f_i^{up}}{\partial m_{j2}^i} = \frac{\partial f_i^{up}}{\partial \mu_{ij}^{up}} \frac{\partial \mu_{ij}^{up}}{\partial m_{j2}^i} = \begin{cases} f_i^{up} \frac{x_j - m_{j2}^i}{(\sigma_j^i)^2} & x_j > m_{j2}^i \\ 0 & \text{otherwise} \end{cases} \quad (30)$$

$$\frac{\partial f_i^{lo}}{\partial m_{j2}^i} = \frac{\partial f_i^{lo}}{\partial \mu_{ij}^{lo}} \frac{\partial \mu_{ij}^{lo}}{\partial m_{j2}^i} = \begin{cases} f_i^{lo} \frac{x_j - m_{j2}^i}{(\sigma_j^i)^2} & x_j \leq \frac{m_{j1}^i + m_{j2}^i}{2} \\ 0 & \text{otherwise} \end{cases} \quad (31)$$

$$\frac{\partial f_i^{up}}{\partial \sigma_j^i} = \frac{\partial f_i^{up}}{\partial \mu_{ij}^{up}} \frac{\partial \mu_{ij}^{up}}{\partial \sigma_j^i} = \begin{cases} f_i^{up} \frac{(x_j - m_{j1}^i)^2}{(\sigma_j^i)^3} & x_j < m_{j1}^i \\ f_i^{up} \frac{(x_j - m_{j2}^i)^2}{(\sigma_j^i)^3} & x_j < m_{j2}^i \\ 0 & \text{otherwise} \end{cases} \quad (32)$$

$$\frac{\partial f_i^{lo}}{\partial \sigma_j^i} = \frac{\partial f_i^{lo}}{\partial \mu_{ij}^{lo}} \frac{\partial \mu_{ij}^{lo}}{\partial \sigma_j^i} = \begin{cases} f_i^{lo} \frac{(x_j - m_{j2}^i)^2}{(\sigma_j^i)^3} & x_j \leq \frac{m_{j1}^i + m_{j2}^i}{2} \\ f_i^{lo} \frac{(x_j - m_{j1}^i)^2}{(\sigma_j^i)^3} & x_j > \frac{m_{j1}^i + m_{j2}^i}{2} \end{cases} \quad (33)$$

### 3.3. Sample reserve strategy

The sample is reserved for training in the later stage if it does not satisfy all the conditions. As mentioned earlier in the sample update section, the samples are only learnt if the prediction error is above the update threshold and are discarded if it is smaller. The sample is reserved for later processing if its prediction error is not only small enough to be deleted but not big enough to update the rule in the system.

#### 4. Performance evaluation

The performance evaluation of McIT2FIS is tested on the wave data collected from the buoys located in Singapore. The data was sampled every half an hour during the period from October 2014 to September 2015. In this work, samples from the first month are studied. 1200 samples are for training and 300 for testing the network. The performance of the algorithm has been evaluated in Matlab R2013b environment on a Window system with Xeon CPU and 16GB RAM.

The actual and predicted output for the first five hundred training wave heights are shown in Figure 2. As can be easily seen, McIT2FIS is able to approximate the function and accurately fit the training data into the model. Figure 3 describes the result for testing data. It can be seen from the figures that the algorithm well generalizes the underlying trend.

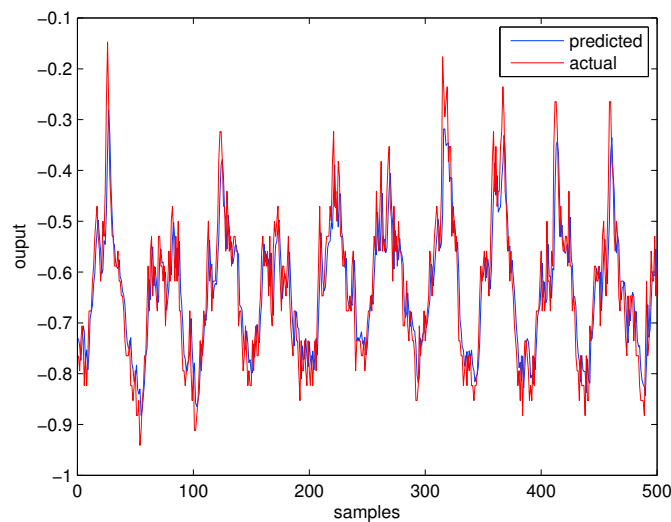


Fig. 2: Actual and predicted wave height for train data.

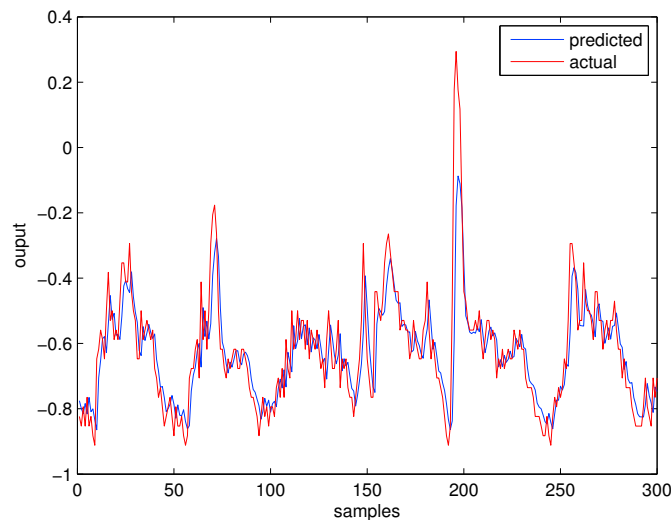


Fig. 3: Actual and predicted wave height for test data.

Now, we compare the performance of the proposed McIT2FIS with other algorithms in the literature based on the number of rules employed and root mean square error. Table I shows the results for the algorithms in this study. It can be observed that McIT2FIS needs smaller number of rules compared to McIT2FIS-GD and acquires competitive prediction.

Table 1: Performance comparison on wave prediction problem

Algorithm	Rules	Training RMSE	Testing RMSE
ANFIS	2	0.06	0.09
McIT2FIS-GD	4	0.159	0.168
<b>McIT2FIS</b>	<b>2</b>	<b>0.066</b>	<b>0.097</b>

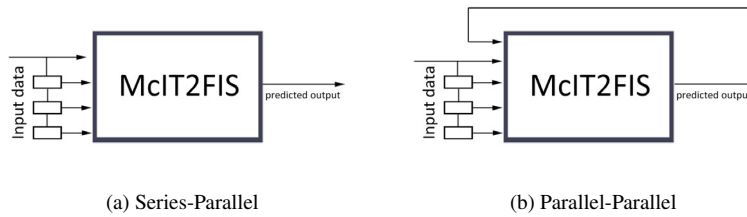


Fig. 4: Series-Parallel and Parallel-Parallel model

Next, we conduct the study on long-term forecasting. In order for McIT2FIS to identify longer step ahead, Series-Parallel and Parallel-Parallel models are employed [14]. In the Series-Parallel model, the output is fed back into the model while with Parallel-Parallel model, the predicted output is used to re-train McIT2FIS. Figure 4 shows the model of two motivating ideas. Performance of two-step ahead forecast using Series-Parallel and Parallel-Parallel model is shown in Figures 5 and Figure 6 respectively. It could be noticed that the two models achieve comparable prediction. Table II gives the root mean square error of the testing data set for both the algorithms when we attempt to forecast longer step ahead. From the table, it could be observed that Series-Parallel model attains slightly smaller error than Parallel-Parallel model. It should also be noted that the prediction error increases fast when we forecast the wave height in longer-term.

Table 2: Performance comparison on wave prediction problem

Testing RMSE	2-step ahead	3-step ahead	4-step ahead	5-step ahead
Series-Parallel	0.1306	0.1505	0.1630	0.1688
Parallel-Parallel	0.1314	0.1502	0.1625	0.1685

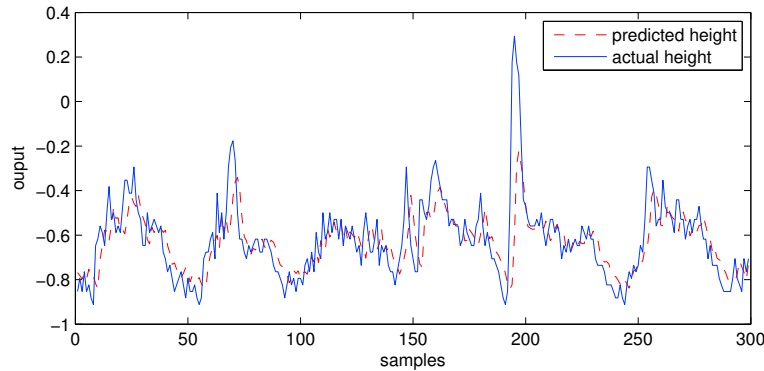


Fig. 5: Two step ahead forecasting using Series-Parallel model

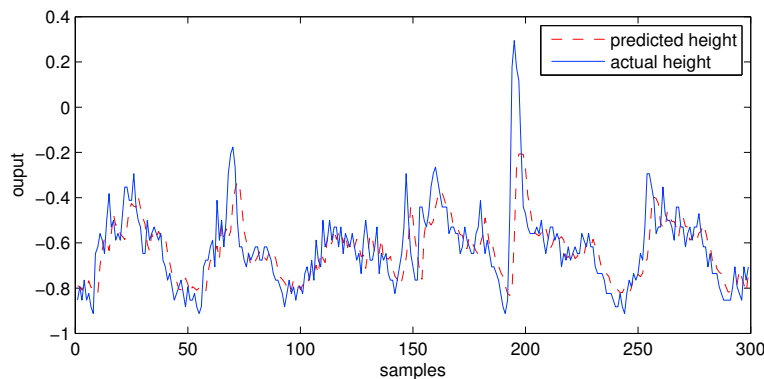


Fig. 6: Two step ahead forecasting using Parallel-Parallel model

## 5. Conclusion

In this study, an interval type-2 fuzzy inference system with its meta-cognitive learning algorithm is evaluated on the wave rider data set. Meta-cognitive component assesses the knowledge presenting in each sample to decide what-to-learn, how-to-learn and when-to learn effectively. The sample is learnt in case the knowledge present is novel to the system while it is deleted when the same knowledge has been there. Learning strategies are based on the novelty criterion as spherical potential and energy function criterion as hinge loss error. Based on this tactics, it is ensure that the redundant computation is avoided and the algorithm runs efficiently. The performance is evaluated on a significant wave height prediction problem and it has been shown that the algorithm can generalize the trend accurately.

## Acknowledgements

The present research was supported from EDB's Energy Innovation Research (EIRP) grant S14-1187-NRFEIPO-EIRP-IHL and the first author's PhD scholarship was generously supported by the IGS School of Nanyang Technological University.

## References

- [1] Abiyev, R.H., Kaynak, O., 2010. Type 2 fuzzy neural structure for identification and control of time-varying plants. *IEEE Transactions on Industrial Electronics* 57, 4147–4159. doi:[10.1109/TIE.2010.2043036](https://doi.org/10.1109/TIE.2010.2043036).
- [2] Abraham, A., 2005. *Adaptation of Fuzzy Inference System Using Neural Learning*. Springer Berlin Heidelberg, Berlin, Heidelberg.

- [3] Babu, G.S., Suresh, S., 2012. Meta-cognitive neural network for classification problems in a sequential learning framework. *Neurocomputing* 81, 86 – 96.
- [4] Babu, G.S., Suresh, S., 2013. Meta-cognitive rbf network and its projection based learning algorithm for classification problems. *Applied Soft Computing* 13, 654 – 666.
- [5] Castro, J.R., Castillo, O., Melin, P., Rodríguez-Díaz, A., 2009. A hybrid learning algorithm for a class of interval type-2 fuzzy neural networks. *Information Sciences* 179, 2175 – 2193. Special Section on High Order Fuzzy Sets.
- [6] Das, A.K., Anh, N., Suresh, S., Srikanth, N., 2016. An interval type-2 fuzzy inference system and its meta-cognitive learning algorithm. *Evolving Systems* 7, 95–105. doi:[10.1007/s12530-016-9148-6](https://doi.org/10.1007/s12530-016-9148-6).
- [7] Das, A.K., Subramanian, K., Sundaram, S., 2015. An evolving interval type-2 neurofuzzy inference system and its metacognitive sequential learning algorithm. *IEEE Transactions on Fuzzy Systems* 23, 2080–2093. doi:[10.1109/TFUZZ.2015.2403793](https://doi.org/10.1109/TFUZZ.2015.2403793).
- [8] Das, A.K., Subramanian, K., Suresh, S., 2014. A computationally fast interval type-2 neuro-fuzzy inference system and its meta-cognitive projection based learning algorithm, in: 2014 International Joint Conference on Neural Networks (IJCNN), pp. 1510–1516. doi:[10.1109/IJCNN.2014.6889610](https://doi.org/10.1109/IJCNN.2014.6889610).
- [9] Hashim, M.A., Jaafar, J., Taib, S.M., 2012. Interval type 2 fuzzy logic for significant wave height forecasting, in: Proc. 2012 12th International Conference on Intelligent Systems Design and Applications (ISDA), pp. 794–798. doi:[10.1109/ISDA.2012.6416638](https://doi.org/10.1109/ISDA.2012.6416638).
- [10] Karnik, N.N., Mendel, J.M., Liang, Q., 1999. Type-2 fuzzy logic systems. *IEEE Transactions on Fuzzy Systems* 7, 643–658. doi:[10.1109/91.811231](https://doi.org/10.1109/91.811231).
- [11] Kosko, B., 1992. *Neural Networks and Fuzzy Systems: A Dynamical Systems Approach to Machine Intelligence*. Prentice-Hall, Inc., Upper Saddle River, NJ, USA.
- [12] Liang, Q., Mendel, J.M., 2000. Interval type-2 fuzzy logic systems: theory and design. *IEEE Transactions on Fuzzy Systems* 8, 535–550. doi:[10.1109/91.873577](https://doi.org/10.1109/91.873577).
- [13] 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, 959–969. doi:[10.1109/TNNLS.2013.2284603](https://doi.org/10.1109/TNNLS.2013.2284603).
- [14] Narendra, K.S., Parthasarathy, K., 1990. Identification and control of dynamical systems using neural networks. *IEEE Transactions on Neural Networks* 1, 4–27. doi:[10.1109/72.80202](https://doi.org/10.1109/72.80202).
- [15] Özger, M., 2010. Significant wave height forecasting using wavelet fuzzy logic approach. *Ocean Engineering* 37, 1443 – 1451. doi:[10.1016/j.oceaneng.2010.07.009](https://doi.org/10.1016/j.oceaneng.2010.07.009).
- [16] Park, K.J., Oh, S.K., Pedrycz, W., 2009. Design of interval type-2 fuzzy neural networks and their optimization using real-coded genetic algorithms, in: 2009 IEEE International Conference on Fuzzy Systems, pp. 2013–2018.
- [17] Pratama, M., Zhang, G., Er, M.J., Anavatti, S., 2016. An incremental type-2 meta-cognitive extreme learning machine. *IEEE Transactions on Cybernetics* 47, 339 – 353.
- [18] Savitha, R., Suresh, S., Sundararajan, N., 2012. A meta-cognitive learning algorithm for a fully complex-valued relaxation network. *Neural Networks* 32, 209 – 218. Selected Papers from IJCNN 2011.
- [19] Subramanian, K., Savitha, R., Suresh, S., 2014. A complex-valued neuro-fuzzy inference system and its learning mechanism. *Neurocomputing* 123, 110 – 120. Contains Special issue articles: Advances in Pattern Recognition Applications and Methods.
- [20] Subramanian, K., Suresh, S., 2012. A meta-cognitive sequential learning algorithm for neuro-fuzzy inference system. *Applied Soft Computing* 12, 3603 – 3614. doi:[10.1016/j.asoc.2012.06.012](https://doi.org/10.1016/j.asoc.2012.06.012).