# INTERVAL TYPE-2 FUZZY NEURAL NETWORKS FOR SHORT-TERM ELECTRIC LOAD FORECASTING:A COMPARATIVE STUDY

Uduak Umoh, Ini Umoeka, Mfon Ntekop and Emmanuel Babalola

Department of Computer Science, University of Uyo, Akwa Ibom State, Nigeria

### *ABSTRACT*

*This paper focuses on the study of short term load forecasting (STELF) using interval Type-2 Fuzzy Logic (IT2FL) and feed-forward Neural Network with back-propagation (NN-BP) tuning algorithm to improve their approximation capability, flexibility and adaptiveness. IT2FL for STELF is presented which provides additional degrees of freedom for handling more uncertainties for improving prediction accuracy and reducing cost. The IT2FL comprises five components which include; the fuzzification unit, the knowledge base, the inference engine, the type reducer and the defuzzification unit. Gaussian membership function is used to show the degree of membership of the input variables. The lower and upper membership functions (fired rules) as well as their consequent coefficients of IT2FL are fed into a (NN) which produces a crisp value coresponding to the optimal defuzzified output of IT2FLSs. The NN type reducer is trained to optimize parameters of membership function (MF) so as to produce an output with minimum error function with the purpose of improving forecasting performance of IT2FLS models. The IT2FNN system has the ability to overcome the limitations of individual technique and enhances their strengths to handle electric load forecasting. The IT2FNN is applied for STELF in Akwa Ibom State-Nigeria. The result of performance of IT2FNN is compared with IT2FLS and T1FLS methods for short term load forecasting with MSE of 0.00123, 0.00185 and 0.00247 respectively. Also, the results of forecasting are compared using RMSE of 0.035, 0.043 and 0.035 respectively, indicating a best accurate forecasting with IT2FNN. In addition, the result of performance of IT2FNN is compared with IT2FLS and T1FLS methods for short term load forecasting with MAPE of 1.5%, 3% and 4.5% respectively. Simulation results show that the IT2FNN approach takes advantages of accuracy and efficiency and performs better in prediction than IT2FL and T1FL methods in power load forecasting task. .*

### *KEYWORDS*

*Interval type-2 fuzzy logic; feed-forward neural network; back propagation; electric load; Interval type-2 fuzzy neural networks; type-1 fuzzy logic.*

## 1. INTRODUCTION

Akwa Ibom State in Nigeria has an energy demand that is continuously rising due to its increasing population. The role that affordable and reliable electricity plays in shaping the world economy cannot be overemphasized, as a nation's growth in the gross domestic product (GDP) can be trailed to its growth in electricity. Accurate estimation of future power demands is required to facilitate the task of generating power reliably and economically. Due to the growing rate of residential, industrial, commercial, economic development and increase in population, clean, constant and efficient electric power supply is needed for the rapid growth and development of any given society, especially in a developing economy as Nigeria. It is observed that the electric power supplied in Nigeria is not adequate and cannot meet the demand needed for residential, commercial and industrial purposes. This has given rise to frequent power failures, fluctuations and outrages, leading to loss of revenue of utility companies, loss of energy utilization by the

DOI:10.5121/ijsc.2018.9101 

customers and extra indirect cost. The consumption of electricity has gained critical attention as it acts as a production factor for corporations and welfare factor for societies.

Electric Load is defined as an electric component or portion of an electric circuit that consumes electric power. The electric load consumed in any given power system depends on the amount and type of electric components being powered. Electric Load Forecasting is defined as the process of predicting or estimating the amount of electricity that will be needed to fully power up residences, industries and other institutions in the future. Electric load forecasting is one of the central functions in any power system to ensure an effective operation and planning of the system. Electric Load forecasting has become essential for efficient power system planning and operation. The forecasts for different time horizons are important for different operations within a utility company. Generally, the predictions are made hourly, or daily, or weekly, or monthly or yearly. There are three kinds of electric load forecasting depending on its time scale. These include; short-, medium- and long-term [1]. Short term electric load forecasts range from one hour to one week. Medium term electric load range from a week to a year, while the Long load forecast are for predictions beyond a year. Accurate short- and medium-load forecasting is still a challenging problem. This is due to nonlinear and random behaviour of load demands. Short-term forecasts have become increasingly important due to extensive rise of the competitive market [2]. In particular, STLF is essential for variety of decision making processes such as expansion planning, transaction evaluation, economic dispatch, operation and system reliability, energy-efficiency of a power system, etc.

The power sector in Nigeria is undergoing various structural and organizational changes in recent past. The main focus of all the changes initiated is to make the power system more efficient, economically viable and better service oriented. All these can happen if, among other vital factors, there is a good and accurate system in place for forecasting the load that would be in demand by electricity customers. Such forecasts will be highly useful in proper system planning and operations.

Electric load forecast, being a non-stationary random process, is affected by many factors including but not limited to weather conditions, social and economic environment, electricity price amendments, and calendar information (regular workdays and anomalous days), customer classes, etc. As the electricity market deepens reform, uncertainties inherent in the power system make it so difficult to predict power load accurately. Whereas, having an accurate prediction of electric load is very important for several reasons including the economy and security. So finding an actual approach for handling uncertainty in electric load forecasting for improving the accuracy is very important. High accuracy of the load forecasting can increase network reliability, improve the security of the power system, reduces rates of equipment failures, and reduces the costs and blackouts

Different methods are proposed for the short-term forecasting of electricity load, ranging from statistical models to more complicated artificial and computational intelligence. Statistical forecasting methods have been widely applied for load forecasting because of their simplicity and low computational cost [3] [4]. However, due to their linearity, their usage in describing the nonlinear and seasonal patterns of loads is significantly restricted. Also, advanced nonlinear methods from the field of artificial intelligence have also been widely utilized for electric load forecasting. These models are divided into several sub-groups such as neural networks (FNNs), fuzzy logic systems, support vector machines, evolutionary computing, hybrid and other approaches [5 - 10].

However, all of these models present remarkable records in a certain situations while simultaneously fail in others and, moreover, each possibility offers different information and precision in handling load forecasting. Also, since STLF behaviour are non-linear and influenced by many factors including imprecision, uncertainty and vagueness , some of these techniques are not suitable for analyzing load with highly non-linear characteristics like, weather condition and some varying activities [11 - 13].

Recently, Fuzzy logic and neural networks are widely applied to solve real world problems. Fuzzy logic is a set of mathematical principles for knowledge representation based on degrees of membership rather than the classical binary logic. It is a powerful tool to tackle imprecision and uncertainty and was initially introduced to improve robustness and low-cost solutions for real world problems [14]. Generally, the type-1 fuzzy logic systems have been implemented in many systems to a wider scale some of which include approximation and forecasting systems, control systems, databases, healthcare clinical diagnosis and so on.

The drawback of the conventional fuzzy logic (type-1) is in its limited capabilities to directly handle data uncertainties as some of the designed systems face high level of uncertainties that can affect the performances of the systems. The type-2 fuzzy logic system (T2FLS) is an extension of the former with the intention of being able to model the uncertainties that invariably exist in the rule base because the membership functions of type-2 fuzzy systems are themselves fuzzy. They provide a powerful framework to represent and handle such types of uncertainties. An interval type-2 fuzzy logic system (IT2FLS), which is a special case of T2FLS, has been applied to solve real-world problems. Recent theoretical and practical studies confirm that IT2FLSs cope well in handling uncertainties adequately than their type-1 (T1) counterparts and it is reasonably expected to witness more and more applications of IT2FLSs in different fields of science and engineering. The T1FLS and IT2FLS have been applied to solve problems in a wide variety of areas, including STELF [15 - 22].

Neural network (NN) models in artificial intelligence are usually referred to as artificial neural networks (ANNs). These are essentially simple mathematical models defining a function $f: X \rightarrow Y$ or a distribution over $X$ or both. Sometimes NN models are also intimately associated with a particular learning algorithm or learning rule. An ANN is typically defined by three types of parameters: (1) the interconnection pattern between the different layers of neurons. (2) The learning process for updating the weights of the interconnections. (3) The activation function that converts a neuron's weighted input to its output activation. NN have the ability of learning nonlinear relationships in a system, because of dealing with nonlinear patterns, NN has shown better performance in accuracy in contrast to the traditional statistical methods such as discriminant analysis and logistic regression.

Hybrid systems combining fuzzy logic, neural networks, genetic algorithms, and expert systems are proving their effectiveness in a wide variety of real world problems. Every intelligent technique has particular computational properties (e.g. ability to learn, explanation of decisions) that make them suited for particular problems and not for others. For example, while neural networks are good at recognizing patterns, they are not good at explaining how they reach their decisions. Fuzzy logic systems, which can reason with imprecise information, are good at explaining their decisions but they cannot automatically acquire the rules they use to make those decisions. These limitations have been a central driving force behind the creation of intelligent hybrid systems where two or more techniques are combined in a manner that overcomes the limitations of individual techniques. Type-2 fuzzy neural systems have been developed and applied to several data mining problems. Recent studies on load forecasting report that IT2FLSs possess an excellent approximation capability even better than traditional nonparametric methods such as NNs. But the integrated IT2FLS and NN have self-learning characteristics and allow reducing the complexity of the data and modelling uncertainty and imprecision [23 - 33].

In this paper, we study IT2FL and combine mainly IT2FL and NN approaches to improve their approximation capability, flexibility and adaptiveness for improving power load forecasting accuracy. IT2FL for STELF is presented which provides additional degrees of freedom for handling more uncertainties for improving prediction accuracy and reducing cost. The IT2FL comprises five components which include; the fuzzification unit, the knowledge base, the inference engine, the type reducer and the defuzzification unit. Gaussian membership function is used to show the degree of membership of the input variables. The lower and upper membership functions (fired rules) as well as their consequent coefficients of IT2FL are fed into a (NN) which produces a crisp value coresponding to the optimal defuzzified output of IT2FLSs. The NN type reducer is trained to optimize parameters of membership function (MF) so as to produce an output with minimum error function with the purpose of improving forecasting performance of IT2FLS models. The IT2FNN system has the ability to overcome the limitations of individual technique and enhances their strengths to handle electric load forecasting.

The study considers information such as temperature, humidity and the past electric load data. A study case of Uyo, Akwa Ibom State in Nigeria is considered, where all weather data are obtained from the Nigeria Meteorological Agency (NMA) and historical electric load data are obtained from the Power Holding Company of Nigeria (PHCN). The result of performance of IT2FNN is compared with IT2FLS and T1FLS methods for short term load forecasting with MSE of 0.00123, 0.00185 and 0.00247 respectively. Also, the results of forecasting are compared using RMSE of 0.035, 0.043 and 0.035 respectively, indicating a best accurate forecasting with IT2FNN. In addition, the result of performance of IT2FNN is compared with IT2FLS and T1FLS methods for short term load forecasting with MAPE of 1.5%, 3% and 4.5% respectively. Simulation results show that the IT2FNN approach takes advantages of accuracy and efficiency and performs better in prediction than IT2FL and T1FL methods in power load forecasting task.
The remainder of the paper is presented as follows; section two gives IT2FL overview. In section three, IT2FL for STELF is presented. Section four presents IT2FNN for STELF, the simulation results are shown in section and conclusion is performed in section six.

## 2. INTERVAL TYPE-2 FUZZY LOGIC: AN OVERVIEW

### 2.1. INTERVAL TYPE-2 FUZZY SETS

An interval type-2 fuzzy set (IT2FS) $\tilde{A}$ is characterized as:

$$\tilde{A} = \left\{\left((x, u), \mu_{\tilde{A}}(x, u)\right) \middle| \forall\, x \in X, \forall\, u \in J_x \subseteq [0, 1]\right\} \quad (1)$$

where *x*, the *primary variable*, has domain *X*; *u*∈*U*, the *secondary variable*, has domain $J_x$ at each *x*∈*X;* $J_x$ is called the primary membership of x and the secondary grades of $\tilde{A}$ all equal 1. Uncertainty about $\tilde{A}$ is conveyed by the union of all the primary memberships, which is the shaded region bounded by upper membership function (UMF) and lower membership functions (LMF) is called the *footprint of uncertainty* (FOU) of $\tilde{A}$ as shown in Figure 1 [34] [35] [36].

$$\mu_{\tilde{A}}(x, u) = 1, FOU\,(\tilde{A}) = \cup_{\forall x \in X} J_x = \{(x, u) : u \in Jx \subseteq [0, 1]\} \quad (2)$$

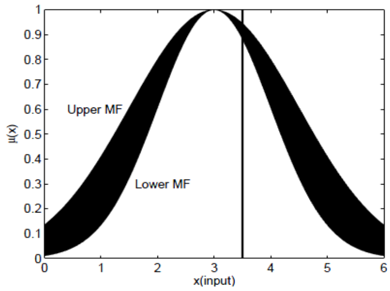


Fig. 1: Interval Type-2 Fuzzy set with uncertain mean [24]

$$\text{UMF} = \overline{\mu}_{\tilde{A}}(x) \equiv \overline{FOU\ (\tilde{A})} \quad \forall x \in X \tag{3}$$

$$\text{LMF} = \underline{\mu}_{\tilde{A}}(x) \equiv \underline{FOU\ (\tilde{A})} \ \ \forall x \in X \tag{4}$$

$$J_x = \{(x, u) : u \in [\underline{\mu}_{\tilde{A}}(x), \overline{\mu}_{\tilde{A}}(x)]\} \tag{5}$$

Where, $\overline{\mu}_{\tilde{A}}(x)$ and $\underline{\mu}_{\tilde{A}}(x)$ of $\tilde{A}$, are two type-1 MFs that bound the FOU, $J_x$ is an interval set. Set theory operations of union, intersection and complement can be applied to easily compute for IT2 FSs.

## 2.2. Interval Type-2 Fuzzy Logic System (IT2FLS)

Figure 1 gives a typical structure of IT2FLS. The components of IT2FLS are similar to T1FLS with the exception of the type reduction (TR) unit. The structure of IT2FLS is made up of; a fuzzifier, an if–then rule base, inference engine, a type-reducer and a defuzzifier.

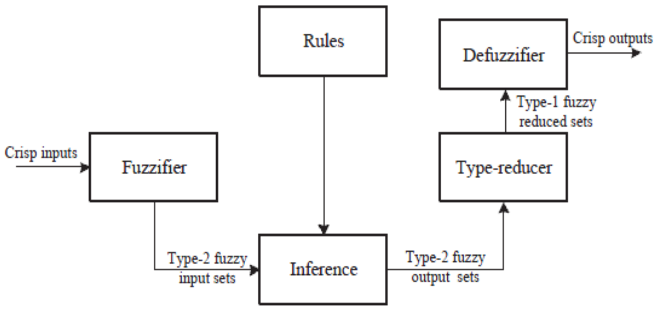


Fig. 2: The Structure of IT2FLS [37].

Number The fuzzification process maps a crisp input vector into type-1 or IT2FSs using singleton, triangular, trapezoidal or Gaussian fuzzifier. Input IT2-FSs then activate the inference engine and the rule base to produce output IT2 FSs. A type-reduction unit combines the output sets and then performs a centroid calculation which produces an interval T1FS (type-reduced set). The type-reduced set is then processed by the defuzzification unit to produce crisp outputs.

In this paper, IT2 Gaussian MFs with uncertain width (deviation) is used because of it suitability for highly dynamic random problem, such as electric load forecasting.

$$f(x) = exp\left(-\frac{x-c}{2\sigma^2}\right), \sigma \in [\sigma_1, \sigma_2] \text{ and } c \in [c_1, c_2] \tag{6}$$

That is $\sigma \in [\sigma_1, \sigma_2]$ with upper and lower membership functions defined as follows:

$$\bar{\mu}_{\tilde{A}_{im}}(x_i) = \exp\left(-\frac{x_i - c_{im}}{2\bar{\sigma}_{2,im}^2}\right),\ \bar{\mu}_{\tilde{A}}(x) = N(c, \sigma_2; x) \quad (7)$$

$$\underline{\mu}_{\tilde{A}_{im}}(x_i) = \exp\left(-\frac{x_i - c_{im}}{2\underline{\sigma}_{1,im}^2}\right),\ \underline{\mu}_{\tilde{A}}(x) = N(c, \sigma_1; x) \quad (8)$$

Where c is the center (mean) of the MF, σ is the width (standard deviation) of the MF and *x* is the input vector. The variables $\bar{\sigma}_{2,im}$ and $\underline{\sigma}_{1,im}$ are premise parameters that define the degree of membership of each element to the fuzzy set Ã and FOUs of the IT2IFS. The MFs are defined and evaluated for all the input and output linguistic variables. The IT2F sets are explored in the antecedents' parts and each MF of the antecedent part is represented using an upper and a lower MFs, denoted by $\bar{\mu}_{\tilde{A}}$(x) and $\underline{\mu}_{\tilde{A}}$(x). The detail description is found in [24]. Each node output indicates the lower and upper interval.

An IT2FLS is characterized by if–then rules which can be expressed as a collection of IF–THEN statements as;

$$R^n: IF\ x_i\ is\ \tilde{F}_i^l\ AND, \ldots, AND\ x_m\ is\ \tilde{F}_m^n\ THEN\ y\ is\ \tilde{F}^n; Y^n = [\underline{y}^n, \bar{y}^n] \quad (9)$$

Where $x_i$ , *i = 1, m* are the antecedents, $y$ is the consequent of the *ith* rule of IT2FLS. The $\tilde{F}^i$'s are the MFs $\mu_{\tilde{F}_i^l}(x_i)$ of the antecedent part assigned of the *i*th input $x_i$ , the $F^l$ is the MFs $\mu_{\tilde{E}_j^l}(y)$ of the consequent part assigned to the output $y_j$. Also, $\underline{y}^n$ and $\bar{y}^n$ are the lower and upper coefficients of the consequent part and $Y^n$ is the IT1 set corresponding to the centroid of the IT2 consequent set. The result of the input and antecedent operations contained in the firing set produces an interval type-1 set as shown in (10) [38].

$$F^i(x') = [\underline{f'}(x'), \overline{f'}(x')] \equiv [\underline{f'}, \overline{f'}] \quad (10)$$

$$\underline{f}^i(x') = \underline{\mu}_{\tilde{f}_1^i}(x_1') * \ldots \ldots * \underline{\mu}_{\tilde{f}_m^i}(x_1') \quad (11)$$

$$\bar{f}^i(x') = \bar{\mu}_{\tilde{f}_1^i}(x_1') * \ldots \ldots * \bar{\mu}_{\tilde{f}_m^i}(x_1') \quad (12)$$

where $F^i(x')$ is the antecedent of rule *i* and $\mu_{F1}{}^i(x')$ is the degree of membership of *x* in F. $\bar{\mu}_{\tilde{f}^i}$(x) and $\underline{\mu}_{\tilde{f}^i}$(x) are upper and lower MFs of $\mu_{\tilde{f}^i}$, *i* = 1 to *m* respectively and * is a t-norm (minimum or product).

The inference engine combines the fired rules and gives a mapping from input to output in IT2FSs. The combined output fuzzy set, $\mu_{\tilde{F}_j^l}(y_j)$ , is obtained by combining the fired output consequent sets by taking the union of the *ith rule* fired output consequent sets. The singleton fuzzifier is applied to obtain (11) and (12).

The output of the IT2FLS model is obtained through combining the outcomes of *N* rules through ttype-reduction process using iterative Karnik-Mendel (KM) algorithm where [39 - 40].

$$Y_{cos}(x') = \bigcup_{\substack{f^n \in F^n(x') \\ y^n \in Y^n}} \frac{\sum_{n=1}^{N} f^n y^n}{\sum_{n=1}^{N} f^n} = [y_l, y_r] \quad (13)$$

Where $f^n \in F^n$ and $y^n \in Y^n$ for n = 1,…., N.

$$y_l = \frac{\sum_{n=1}^{L} \bar{f}^n \underline{y}^n + \sum_{n=L+1}^{N} \underline{f}^n \underline{y}}{\sum_{n=1}^{L} \bar{f}^n + \sum_{n=L+1}^{N} \underline{f}^n} \tag{14}$$

$$y_r = \frac{\sum_{n=1}^{R} \underline{f}^n \bar{y}^n + \sum_{n=R+1}^{N} \bar{f}^n \bar{y}^n}{\sum_{n=1}^{R} \underline{f}^n + \sum_{n=R+1}^{N} \bar{f}^n} \tag{15}$$

Where $y_l$ is the leftmost point and $y_r$ is the rightmost point, L and R are the switch points determined by (14) and (15) respectively. $\underline{y}^L \leq y_l \leq \underline{y}^{L+1}$ , $\bar{y}^R \leq y_r \leq \bar{y}^{R+1}$, $\underline{y}^n$ and $\bar{y}^n$ are arranged in ascending order.

Finally, the defuzzified crisp output IT2FLS is achieved by averaging of $y_l$ and$y_r$, as in (16):

$$Y = \frac{y_l + y_r}{2} \tag{16}$$

In an IT2FLS type reduction process converts an IT2 fuzzy set (FS) into aT1 FS. The first TR algorithm was proposed by Karnik and Mendel (KM) which recursively computes the left and right end points (center of sets) producing a T1 interval set. However, the KM algorithm has computational challenge when there are many MFs and the rule base is large, and may not be suitable for fast real-time applications. In order to solve this problem, many more approached have been proposed [41 - 38]. Despite all these approaches, however, IT2FLS models using these TR algorithms are still computationally more intensive than T1 FLSs. It is found that the UB algorithm generates the closest outputs to the KM TR algorithm [45 - 48]. Previous studies mainly attempt to minimize the computational requirement of TR block compared to original KM algorithm, rather than checking the quality and accuracy of defuzzified outputs computed.

## 3. Interval Type-2 Fuzzy Logic For Short Term Electric Load Forecasting

In this section, the model of IT2FLS-STLF is presented in Figure 3, which IT2FLS comprises five components: fuzzification unit, knowledge base, inference engine, type reducer and defuzzification unit. The electric load data used in this work are presented in Tables 1 and 2 respectively.

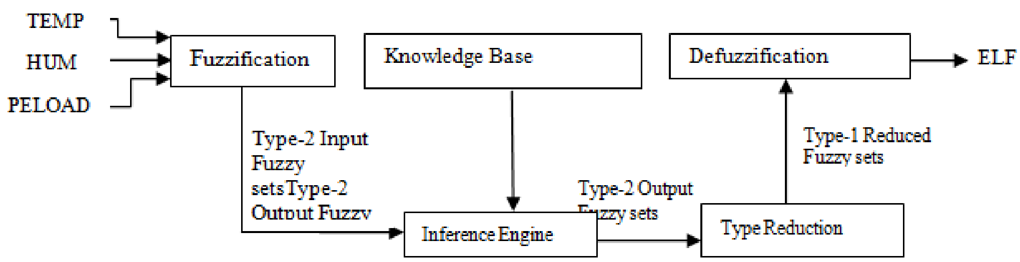


Fig. 3: Model of IT2FLS-STLF

The input variables are defined as temperature (TEMP), humidity (HUM) and past electric load (PELOAD) respectively. The output parameter is defined as electric load forecast (ELF). The input and output linguistic terms are defined as; Temperature (TEMP) = {Very Cool, Cool, Moderate, Hot, Very Hot}, Humidity (HUM) = {Very Low, Low, Moderate, High, Very High}, Past Electric Load (PELOAD) = {Very Low, Low, Moderate, High, Very High} and Electric Load Forecast (ELF) = {Very Low, Low, Moderate, High, Very High}. The input and output space are divided into fuzzy regions (domain intervals) of $X_1$, $X_2$ and, $X_3$ and Y as [23, 34], [45, 98], [18, 35.4] and [10, 100] respectively. The universe of discourse for each input variable is defined in Table 3. IT2Gaussian MFs are defined for all input and output variables of the IT2S-STELF and are presented in Tables 4 –7 respectively. Fuzzy rules are defined to maximize space; part of the IT2FL-STELF rules is presented in Table 8. The firing strength is evaluated based on (10) (11) (12).

Table 1: Electric load data covering 10th – 15th January 2010

| HOUR | MON | TUES | WED | THUR | FRI | SAT | SUN |
|---|---|---|---|---|---|---|---|
| 0:01 | 30.2 | 30.3 | 32.9 | 33.5 | 28.3 | 31.2 | 35.4 |
| 0:02 | 29.8 | 30.5 | 31 | 32 | 28 | 28.4 | 29.8 |
| 0:03 | 29.2 | 30.2 | 28 | 34 | 27.5 | 24.5 | 26.3 |
| 0:04 | 29 | 30 | 28 | 31.4 | 27.4 | 22.8 | 24.2 |
| 0:05 | 28 | 30 | 27.5 | 32.5 | 26.7 | 21.5 | 20.1 |
| 0:06 | 28 | 29 | 27 | 28.9 | 28.4 | 19.5 | 18 |
| 0:07 | 30 | 31.6 | 29.3 | 27.3 | 29.3 | 21.5 | 20.4 |
| 0:08 | 40 | 42 | 37.3 | 32.5 | 32.4 | 23.7 | 23.6 |
| 0:09 | 50.8 | 49.5 | 43 | 34.4 | 35.6 | 25.2 | 23.9 |
| 0:10 | 49.9 | 50.1 | 40 | 45.8 | 37.8 | 38 | 25.1 |
| 11:00 | 49.5 | 50.3 | 39 | 40.2 | 38.9 | 27.5 | 26.7 |
| 12:00 | 49.8 | 50.4 | 40.3 | 39.2 | 39.4 | 27.8 | 26.7 |
| 13:00 | 50.3 | 50.5 | 47.2 | 40.1 | 41.3 | 27.9 | 27.8 |
| 14:00 | 50.8 | 50.7 | 46.9 | 43.8 | 44.2 | 28.4 | 27.4 |
| 15:00 | 50.8 | 50.9 | 45.6 | 44.6 | 45.6 | 28.7 | 27.2 |
| 16:00 | 50.3 | 51 | 46.1 | 48.7 | 46.8 | 29.5 | 26.7 |
| 17:00 | 48.6 | 48.4 | 46.5 | 46.2 | 47.2 | 27.5 | 20.4 |
| 18:00 | 43.8 | 45.5 | 45.8 | 44 | 44.2 | 26.5 | 21.2 |
| 19:00 | 45.7 | 47.6 | 44 | 43.3 | 42.5 | 25.4 | 23.5 |
| 20:00 | 48.7 | 49 | 43.8 | 45.4 | 40.5 | 22.2 | 24.1 |
| 21:00 | 42.4 | 44.5 | 41 | 43.7 | 35.2 | 23.1 | 19.7 |
| 22:00 | 37.5 | 39.5 | 35.6 | 33.2 | 32.1 | 22.4 | 18.2 |
| 23:00 | 33.5 | 35.2 | 32.9 | 30.2 | 30.1 | 21.3 | 18.3 |
| 24:00 | 30.3 | 33.6 | 31.5 | 30.1 | 29.3 | 21 | 18.4 |

Table 2: Electric Load Training Data

| Time (Hr) | Temperature (**0** C) | Humidity (%) | PELoad (MW) |
|---|---|---|---|
| 0:01 | 25 | 93 | 35.4 |
| 0:02 | 25 | 94 | 29.8 |
| 0:03 | 23 | 94 | 26.3 |
| 0:04 | 24 | 97 | 24.2 |
| 0:05 | 24 | 98 | 20.1 |
| 0:06 | 25 | 97 | 18 |
| 0:07 | 24 | 98 | 20.4 |
| 0:08 | 27 | 83 | 23.6 |
| 0:09 | 28 | 72 | 23.9 |
| 0:10 | 30 | 64 | 25.1 |
| 11:00 | 32 | 53 | 26.7 |
| 12:00 | 33 | 48 | 26.7 |
| 13:00 | 34 | 45 | 27.8 |

| | | | |
|---|---|---|---|
| 14:00 | 34 | 46 | 27.4 |
| 15:00 | 33 | 47 | 27.2 |
| 16:00 | 33 | 48 | 26.7 |
| 17:00 | 30 | 54 | 20.4 |
| 18:00 | 29 | 68 | 21.2 |
| 19:00 | 28 | 73 | 23.5 |
| 20:00 | 27 | 80 | 24.1 |
| 21:00 | 26 | 85 | 19.7 |
| 22:00 | 26 | 88 | 18.2 |
| 23:00 | 24 | 90 | 18.3 |
| 24:00 | 23 | 94 | 18.4 |

Table 3: Universe of Discourse for Input Variables

| INPUT VARIABLES AND THEIR UNIVERSE OF DISCOUSE | | |
|---|---|---|
| Temperature (TEMP) | Humidity (HUM) | Past Electric Load (PELOAD) |
| 0 – 50 | 0 – 100 | 0 - 55 |

Table 4: IT2FL-STELF Membership function values for Temperature

| Terms | M (mean) | δ 1(UMF) | δ 2 (LMF) |
|---|---|---|---|
| VCool | 7 | 2 | 1 |
| Cool | 15 | 2.2 | 1.2 |
| Moderate | 25 | 3 | 1.5 |
| Hot | 35 | 2 | 1 |
| VHot | 42 | 2 | 1 |

Table 5: IT2FL-STELF Membership function values for Humidity

| Terms | M (mean) | δ 1(UMF) | δ 2 (LMF) |
|---|---|---|---|
| VLow | 12 | 4 | 2 |
| Low | 28 | 5 | 3 |
| Moderate | 50 | 6 | 4 |
| High | 70 | 5 | 3 |
| VHigh | 8 | 4 | 2 |

Table 6: IT2FL-STELF Membership function values for PELoad

| Terms | M (mean) | δ 1(UMF) | δ 2 (LMF) |
|---|---|---|---|
| VLow | 8 | 2.3 | 1.3 |
| Low | 17 | 2.5 | 1.3 |
| Moderate | 25 | 2.3 | 1 |
| High | 36 | 3 | 1.5 |
| VHigh | 47 | 2.3 | 1.0 |

Table 7: IT2FL-STELF Membership function values for ELF

| Terms | M (mean) | δ 1(UMF) | δ 2 (LMF) |
|---|---|---|---|
| VLow | 8 | 2.3 | 1.0 |
| Low | 16 | 2.8 | 1.1 |
| Moderate | 27 | 2.9 | 1.2 |
| High | 40 | 3 | 1.5 |
| VHigh | 52 | 3 | 1.5 |

Table 8: IT2FL-STELF Fuzzy Rules

| No | Input Variables | | | Output |
|---|---|---|---|---|
| | TEMP | HUM | PELOAD | ELF |
| 1 | VCool | VLow | VLow | VLow |
| 2 | VCool | Low | Low | Low |
| 3 | VCool | Mod | Mod | Mod |
| 4 | Vcool | High | Low | VLow |
| 5 | Cool | High | Mod | Mod |
| 6 | Mod | Mod | Mod | Mod |
| 7 | Mod | Low | Mod | Mod |
| 8 | Mod | Low | Low | Low |
| 9 | Hot | Mod | VLow | Low |
| 10 | Hot | Mod | VHigh | High |
| 11 | Hot | Low | VHigh | High |
| 12 | VHot | Mod | VHigh | VHigh |
| 13 | VHot | Mod | Mod | High |
| .<br>.<br>. | .<br>.<br>. | .<br>.<br>. | .<br>.<br>. | .<br>.<br>. |
| 121 | VHot | VHigh | VLow | VLow |
| 122 | VHot | VHigh | Low | Low |
| 123 | VHot | VHigh | Moderate | Moderate |
| 125 | VHot | VHigh | High | High |
| 125 | VHot | VHigh | VHigh | VHigh |

Interval type-2 inference engine combines the fired rules and gives a mapping from input IT2FSs to output IT2FSs, $\mu_{\tilde{E}_j^i}(y_j)$ for a given input value for of Temp, Hum and Peload. This is achieved by the union of the *ith rule* fired output consequent sets. Type ReductionK-M type-reduction algorithm is applied to compute the left and right end points ($y_l^i$ and $y_r^i$) of the centroid of the consequent of the *i*th rule. $\underline{f}^i$ and $\overline{f}^i$ are the lower and upper firing degrees of the *i*th rule and *N* is the number of fired rules. From the KM algorithms, we find that L = 1 and R = 3 and compute ($y_l^i$ and $y_r^i$). Using our selected sets of input values, we defuzzify the fuzzy set using the average of $y_l$and $y_r$ in and obtain crisp value (Electric Load Forecast, ELF).

## 4. INTERVAL TYPE-2 FUZZY NEURAL SYSTEM (IT2FNNS) FOR SHORT TERM ELECTRIC LOAD FORCASTING

In this section, IT2FNNS for short term electric load forecasting in Uyo, Akwa Ibom State, Nigeria as a study case, is investigated. The study intends to maintain less computational intensiveness and improve the accuracy and capability of IT2FLS in forecasting and approximation by integrating back propagation algorithm of feed-forward NNs. The architecture of IT2FNNS, with basically a five-layer IT2FNNs for short term electric load forecasting is shown in Figure 4. Layer I (Input Layer) provides the input nodes, which are crisp values. Layer II gives IT2 fuzzification nodes which maps the crisp input to a fuzzy set using a defined MF and produces IT2 MFs. They form the antecedent part of this T2FNN. Layer III (fuzzy rule layer), evaluates the rules in a rule base against fuzzy set received from fuzzification to produce yet another fuzzy set. The nodes in this layer consist of the firing strength. Layer III nodes combined with layer IV to produce the consequent parts of the fuzzy rule nodes formed from a classical 2-layer FNN with fuzzy rule nodes and output nodes. The links between layer III and layer IV consist of interval weighting factors which determine the actual outputs of this system. In this paper, IT2FNN type reducer is employed in order to achieve a better forecast accuracy. From Figure 4, the firing strengths of rules and their interval centroids are fed into an NN to directly and optimally generate the deffuzzified output of IT2FLS in place of the traditional TR block. The NN training is performed through minimization of an error-based cost function.

The feeding vector to IT2FNN type reducer is formed from all training samples $n$ as given in (17).

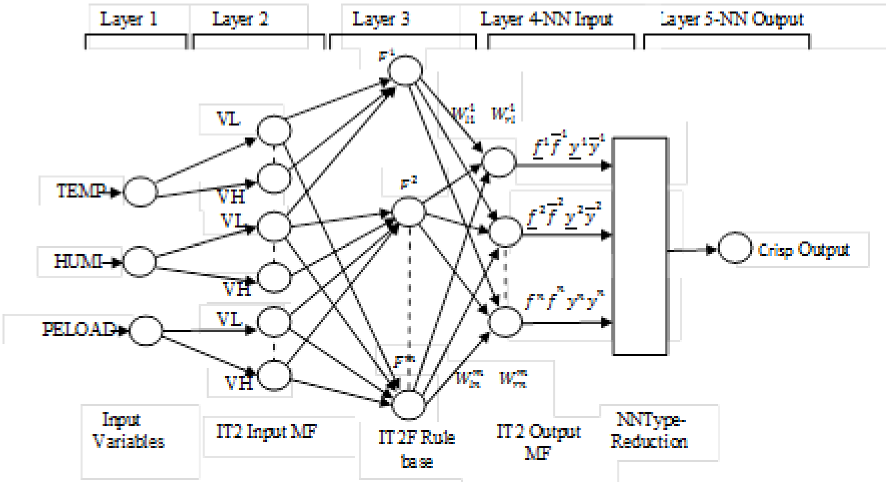


Fig. 4: Architecture of IT2FNNS for STELF

$$w_i = \underline{f}^1, \overline{f}^1, \underline{y}^1, \overline{y}^1, \ldots \underline{f}^n, \overline{f}^n, \underline{y}^n, \overline{y}^n, \; w_i \in \boldsymbol{R}^{4n} \tag{17}$$

Where $w_i$ have the dimension, 1 x 4N, $i = 1, 2, .., n$ and $\underline{f}^n$ and $\overline{f}^n$ give the lower and upper firing strengths of the *nth* rule as given in (11) and (12) respectively. While $\underline{y}^n$ and $\overline{y}^n$ are the centroid of IT2FS in the consequent part of the rules as defined and calculated in (9). The weights of the IT2F output MF values (firing strength), $\{\underline{f}^n\}_{i=1,\ldots n}$ and $\{\overline{f}^n\}_{i=1,\ldots n}$ in (11) and (12), form inputs to the FNN and $w_i$ is computed for each set of input parameters. Using $w_i$ as the input vector, the *i*th desired input–output pattern pair of the new dataset is formed as follows:

$$w_i = \{w_i, T_i\}_{i=1,\ldots n} \tag{18}$$

Thus, a mapping from $w_i \in \boldsymbol{R}^{4n}$ to $T_i \in \mathbf{R}$, i.e., $f(w_i): \boldsymbol{R}^{4n} \rightarrow \mathbf{R}$ is determined and the FNN type reducer is trained using traditional back propagation algorithm [28]. The weights of the IT2F output MF values are tuned using BP algorithm and the error between the desired output $y^K$ and the computed output $O^k$ are determined. During the training process, for each incoming data, the parameters learning optimally tune the parameters of the IT2FNN to find the optimal values. The supervised learning method is used with the objective of minimizing the error function, $E_k$ as defined by means of a learning algorithm.

$$E_k = ½[y^k – O^k]^2 \tag{18}$$

Where, $y^k$ is the desired (expected) output and $O^k$ is the actual (computed) output, k is the input/output pattern. In this study, learning algorithm is adapted and modified from the back BP as described in [49] [50] and is applied to train the IT2FNN.

Finally, the paper explores IT2FNN type reducer with the centers of fuzzy sets and firing strengths of rules to compute the crisp output of an IT2FNN system. Next, the study applies Percentage Error (%Error), Mean Squared Error (MSE), and Root Mean Squared Error (RMSE) and Mean Absolute Percentage Error (MAPE) performance criteria for our experiments which are defined as:

$$\% \, Errr = \frac{|y_i - \bar{y}_i|}{y_i} \times 100 \tag{19}$$

$$MSE = \frac{1}{N}\sum_{i=1}^{N}(y^x - y)^2 \tag{20}$$

$$RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(y^x - y)^2} \tag{21}$$

$$MAPE = \frac{100}{N} \times \sum_{i=1}^{N}\left|\frac{y - y^x}{y}\right| \tag{22}$$

## 5. SIMULATION RESULTS

In this paper, a case study location of Akwa Ibom State in Nigeria is considered, where historical data of past electric load are obtained from the Power Holding Company of Nigeria (PHCN) as well as past weather data from the Nigeria Meteorological Agency (NMA), Uyo in Akwa Ibom State. The principal instrument used for electric load data collection is the daily summary log sheets of the Power Holding Company of Nigeria PHCN. One-month data which covered the period of January 2010 are obtained from the log sheets kept in the undertaking offices of the Power Holding Company of Nigeria (PHCN). The data include the hourly load data transmitted within the Metropolitan city of Uyo. The weather data (Temperature and Relative humidity) for the period of January 2010 for the metropolitan city of Uyo are also obtained from WorldWeatherOnline (http://www.worldweatheronline.com/). Verbal questioning is also carried out.

In order to derive the hourly, daily and weekly load forecast, the study considers factors such as; temperature (TEMP), humidity (HUM) and past electric load (PELOAD). The electric load data use covers 10th – 15th January. The load data from 10 January, 2010 serve as the training data which is used to generate the fuzzy rules for the knowledge base and electric load data for 18 January, 2010 is used for testing purpose. The training data consists of both the input and the output pair. It of the form ($X_1^{(1)}$, $X_2^{(2)}$, $X_3^{(3)}$, $Y^{(1)}$), where $X_1$ (Temp), $X_2$ (Hum) and, $X_3$ (PELoad) are inputs, and Y (ELF) is the output. The performance of the proposed forecasting method depends on the appropriate selection of its set of inputs and its structure.

The results of applying IT2FLS to STEL forecasting on a Monday week is shown in Table 9, while the results of the actual load and the forecasted load (ELF) with % Error is presented in Table 10

Table 9: IT2FL STEL Forecasting Results

| Time | Temperature (°C) | Humidity (%) | PELoad(MW) | ELF(MW) |
|---|---|---|---|---|
| 1am | 24 | 90 | 35.4 | 27.0 |
| 2am | 23 | 94 | 29.8 | 26.9 |
| 3am | 23 | 95 | 26.3 | 24.4 |
| 4am | 24 | 96 | 24.2 | 22.7 |
| 5am | 24 | 98 | 20.1 | 22.1 |
| 6am | 24 | 98 | 18 | 21.3 |
| 7am | 23 | 88 | 20.4 | 19.8 |
| 8am | 24 | 80 | 23.6 | 19.6 |
| 9am | 25 | 74 | 23.9 | 23.9 |
| 10am | 25 | 64 | 25.1 | 23.5 |
| 11am | 24 | 57 | 26.7 | 17.7 |
| 12pm | 26 | 47 | 27.8 | 19.2 |
| 1pm | 28 | 44 | 27.4 | 20.5 |
| 2pm | 29 | 37 | 27.4 | 26.2 |
| 3pm | 30 | 34 | 27.2 | 27.1 |
| 4pm | 31 | 38 | 26.7 | 25.6 |
| 5pm | 30 | 47 | 20.4 | 16.6 |
| 6pm | 29 | 53 | 21.2 | 15.4 |
| 7pm | 29 | 68 | 23.5 | 21.9 |
| 8pm | 28 | 69 | 24.1 | 23.7 |
| 9pm | 26 | 67 | 19.7 | 14.2 |
| 10pm | 25 | 72 | 18.2 | 11.5 |
| 11pm | 24 | 75 | 18.3 | 12.6 |

Table 10: The Results of IT2FL- STEL Forecasting - Load and Actual Load Forecasting with % Error

| Time | Actual Load(MW) | ELF Load (MW) | Error | Error (%) |
|---|---|---|---|---|
| 1am | 26.8 | 27.0 | -0.2 | 0.75 |
| 2am | 25.2 | 26.9 | -1.7 | 6.75 |
| 3am | 23.9 | 24.4 | -0.5 | 2.10 |
| 4am | 22.3 | 22.7 | -0.4 | 1.79 |
| 5am | 21.9 | 22.1 | -0.8 | 0.91 |
| 6am | 22.0 | 21.3 | 0.7 | 3.18 |
| 7am | 20.2 | 19.8 | 0.4 | 1.98 |
| 8am | 20.1 | 19.6 | 0.5 | 2.49 |
| 9am | 24.7 | 23.9 | 0.8 | 3.24 |
| 10am | 24.2 | 23.5 | 0.7 | 2.89 |
| 11am | 18.4 | 17.7 | 0.7 | 3.80 |
| 12pm | 20.1 | 19.2 | 0.9 | 4.48 |
| 1pm | 21.6 | 20.5 | 1.1 | 5.09 |
| 2pm | 25.6 | 26.2 | -1.4 | 2.29 |
| 3pm | 28.2 | 27.1 | 1.1 | 3.90 |
| 4pm | 24.9 | 25.6 | 0.7 | 2.81 |
| 5pm | 17.2 | 16.6 | 0.6 | 3.49 |
| 6pm | 16.1 | 15.4 | 0.7 | 4.35 |
| 7pm | 22.6 | 21.9 | 0.7 | 3.10 |
| 8pm | 22.9 | 23.7 | -0.8 | 3.49 |
| 9pm | 15.0 | 14.2 | 0.8 | 5.33 |
| 10pm | 12.1 | 11.5 | 0.6 | 4.96 |
| 11pm | 11.9 | 12.6 | -0.7 | 5.88 |

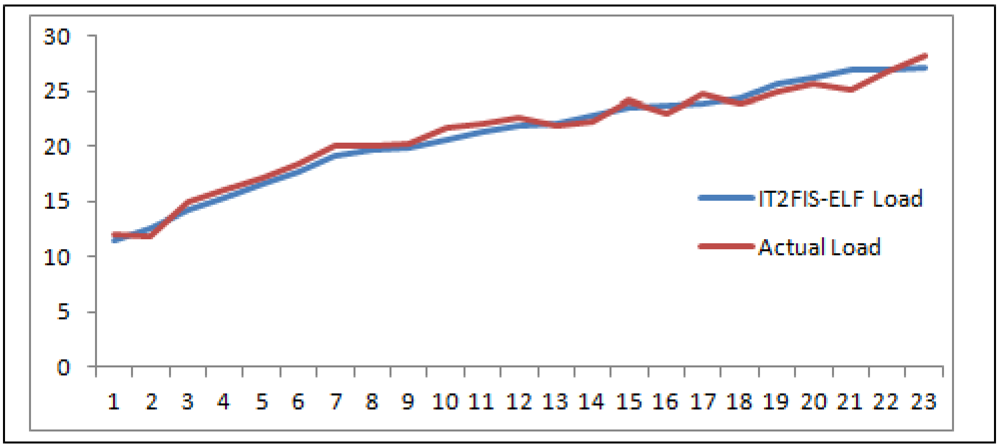


Fig. 5: The curve of Actual Load vs Forecasted Load (ELF)

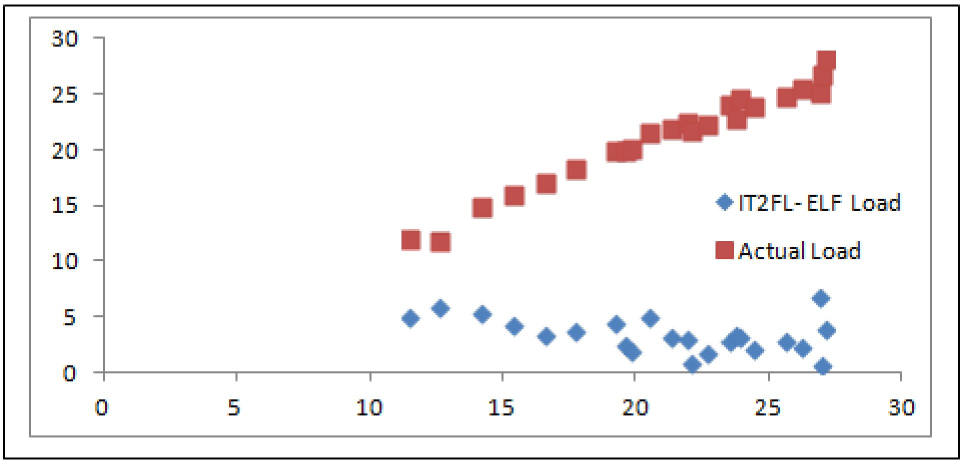


Fig. 6: The plot of Forecasted Load (ELF) and Actual Load against %Error

From IT2FL-STEL forecasting results, it is observed that, the Highest ELE of 27.2 MW is recorded at 30°C Temp, 34% Hum and 27.2MW Peak Load at 1pm and the lowest ELE of 11.5 is achieved at 25°C Temp, 72% Hum and 18.2MW Peak Load at 11pm. The highest and the lowest of forecasting percentage error of hours is at 2am and 1am with 6.75% and 0.75 % respectively, is an over estimation of 2MW. From Table 9, the forecasted ELF load is compared with the actual load with maximum and minimum forecasting percentage error in hours computed at 6.75% and 0.75 %, at 2am and 1am, is an over estimation of 2MW. Figure 5 shows the load curve, plotted based on Table 9, which compares the actual load and the IT2FL forecasted load (ELF). From the curve it is observed that IT2FL forecasted electric load is very close to the actual load with minute variations. This indicates a relatively good correlation with the actual electric load demand. From Figure 6, it is observed that the %error plot against the forecasted and actual electric loads shows that the predicted electric load values indicate a better performance. The results obtained from the IT2FL are compared with the IT1FL method of short term load forecasting.

Parts of the IT2FNN training results is presented in Table 11. Figure 7 presents the graph of Error versus Epoch showing a learning convergence curve for the STELF Training with 0.95 as desired output. In Figure 8, the graph of MSE versus Epoch showing a learning convergence curve for the STELF Training with 0.95 as desired output is shown. In order to comparatively evaluate the performance of the IT2FNN with the T2FL, the results obtained from the IT2FNN are compared with the IT2FLS method for STELF using MSE and RMSE. Table 12 gives parts of the results of comparison of IT2FNN and IT2FLS in STELF. Table 13 gives the comparison of the three approaches based on MSE, RMSE and MAPE performance criteria. Architecture of the graph of IT2FLS versus IT2FNN in STELF is presented in Figure 9. Result indicates that IT2FNN gives the best result with a significant performance improvement over IT2FLS in handling STELF system control.

Table 11: The Results of IT2FNN Training Results for n=5 iterations

| Epochs | Weights | $Y^K$ | $O^K$ | $\delta^o_k$ | MSE |
|---|---|---|---|---|---|
| 1 | [0.100,0.300] [0.150,0.200] | 0.95 | 0.525 | 0.107 | 0.181 |
| 2 | [0.140,0.317] [0.154,0.200] | 0.95 | 0.526 | 0.105 | 0.179 |
| 3 | [0.179,0.319] [0.158, 0.200] | 0.95 | 0.531 | 0.104 | 0.176 |
| 4 | [0.218, 0.335] [0.162, 0.200] | 0.95 | 0.535 | 0.103 | 0.172 |
| 5 | [0.256,0.351] [0.166, 0.200] | 0.95 | 0.539 | 0.102 | 0.169 |

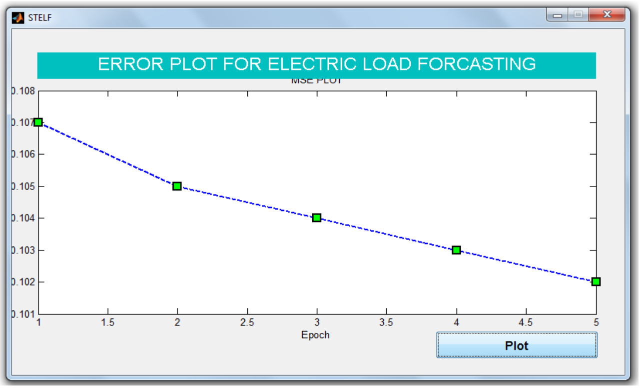


Fig.7: Error versus Epoch showing a Learning Convergence Curve for the STELF Training with 0.95.

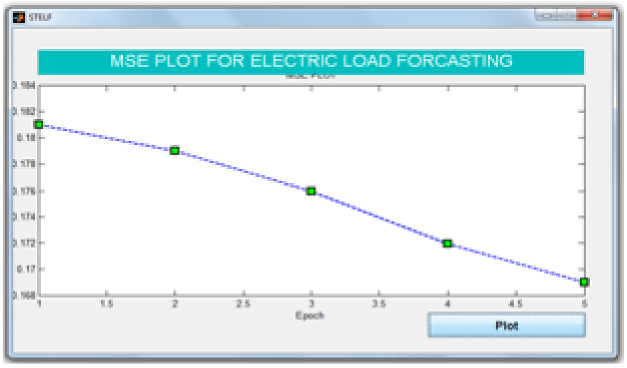


Fig. 8: MSE versus Epoch showing a Learning Convergence Curve for the STELF Training with 0.95.

Table 12: The Results of comparison of IT2FNN and IT2FLS

| Hrs | INPUT | N-WEIG... | N-EPOCH | FNN | DESIRED | FL |
|---|---|---|---|---|---|---|
| 1 | [25, 80, 15] | 2 | 1 | 0.525 | 0.95 | 0.5 |
| 2 | [20.20, 61.65,21.21] | 4 | 46 | 0.5482838745905139 | 0.5357466781311345 | 0.46761504503100754 |
| 3 | [44.87, 7.063,18.07] | 1 | 84 | 0.7049164455589244 | 0.6946104899396027 | 0.6155643669594437 |
| 4 | [15.88, 27.68,38.20] | 4 | 66 | 0.6359785121972443 | 0.6164625124651069 | 0.6159316629204552 |
| 5 | [48.44, 98.00,46.28] | 1 | 84 | 0.6925445728208794 | 0.712391234070994 | 0.6203643082685377 |
| 6 | [11.37, 78.28,31.80] | 2 | 77 | 0.4466312081012732 | 0.4303680328215516 | 0.3873992230863409 |
| 7 | [46.00, 50.63,43.78] | 2 | 72 | 0.44726723191810547 | 0.4642994119794903 | 0.3640021490338934 |
| 8 | [24.16, 68.54,15.46] | 3 | 57 | 0.8697998828647363 | 0.8549906730165554 | 0.7843359625401916 |
| 9 | [4.508, 98.29,39.72] | 4 | 95 | 0.8744409994250195 | 0.8582589032313356 | 0.8152315859819257 |
| 10 | [46.60, 6.799,28.19] | 2 | 52 | 0.9428356973781478 | 0.9267543615583842 | 0.9135684329251417 |
| 11 | [34.39, 90.17,53.59] | 3 | 49 | 0.7140184694385516 | 0.7306119276445148 | 0.6227473945631725 |
| 12 | [11.12, 5.904,30.38] | 1 | 92 | 0.9397261527087293 | 0.9252095735708951 | 0.9332391882930212 |
| 13 | [4.420, 85.48,41.38] | 1 | 85 | 0.7442236885303607 | 0.7325944349717387 | 0.7390704022039494 |
| 14 | [6.653, 71.45,46.57] | 4 | 93 | 0.6308724193013221 | 0.6207709467825128 | 0.5450460557354277 |
| 15 | [43.43, 72.00,54.19] | 2 | 87 | 0.7518377806081529 | 0.7625228822138095 | 0.6695336966247615 |
| 16 | [47.82, 82.55,4.501] | 4 | 85 | 0.5609458309922117 | 0.5440737157413685 | 0.5330353207032209 |
| 17 | [29.28, 80.57,43.53] | 1 | 79 | 0.40267783694251036 | 0.4183493846138743 | 0.3208420608806261 |
| 18 | [37.67, 19.47,11.95] | 3 | 71 | 0.5185887596020877 | 0.5295229401393395 | 0.48925813526872997 |
| 19 | [19.69, 29.83,34.78] | 3 | 70 | 0.9230999943044255 | 0.9066683258860504 | 0.8663080772806 |
| 20 | [40.56, 53.46,35.23] | 4 | 75 | 0.49584186058498886 | 0.5131134744521103 | 0.47119688189609227 |
| 21 | [39.91, 79.82,5.001] | 2 | 83 | 0.46394400802102487 | 0.4761429895906372 | 0.38889848660456294 |
| 22 | [43.35, 41.68,9.214] | 2 | 77 | 0.9995683919600319 | 0.9867097394007969 | 0.9985536050618096 |
| 23 | [11.04, 25.08,47.37] | 4 | 90 | 0.4909701795225908 | 0.4719371580282905 | 0.4208582329472892 |
| 24 | [20.16, 19.14,50.77] | 4 | 68 | 0.433953678192489 | 0.44408587508014347 | 0.4307527861858444 |

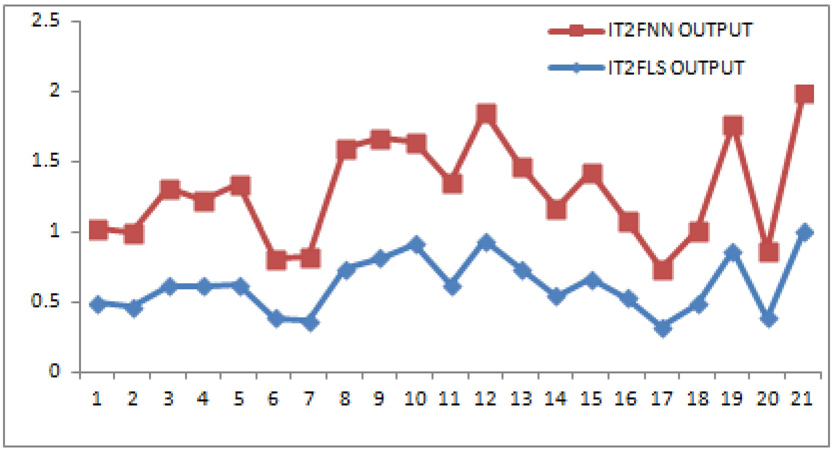


Fig. 9: Architecture of the graph of IT2FLS versus IT2FNN in STELF IT2FNNS for STELF.

Table 13: Comparison of IT2FNN and IT2FLS based on MSE and RMSE

| Models | Training/ Chi | Mean | Standard Deviation | Mean Squared Error (MSE) | Root Mean Square Error (RMSE) | Mean Absolute Percentage Error (MAPE)% |
|---|---|---|---|---|---|---|
| IT1FLS | 300/250 | 0.618 | 0.155 | 0.00247 | 0.050 | 4.5 |
| IT2FLS | 300/250 | 0.626 | 0.198 | 0.00185 | 0.043 | 3 |
| IT2FNN | 400/250 | 0.634 | 0.241 | 0.00123 | 0.035 | 1.5 |

## 5. CONCLUSION

In This paper focuses on the study of short term load forecasting (STELF) using interval Type-2 Fuzzy Logic (IT2FL) and feed-forward Neural Network with back-propagation (NN-BP) tuning algorithm to improve their approximation capability, flexibility and adaptiveness. IT2FLS for STELF is carried out and IT2FNN is introduced for improving performance of IT2FLS models in solving the problem of STELF. Temperature, Humidity and Past Electric Load are used as inputs parameters. By formulating rule base of IT2FLS using available data, outputs (Electric Load Forecast) are obtained with an error margin of 0.75% and 5.88%. The consequent parts of the IT2FLS form an input to NN, where the firing strengths of rules and their interval centroids are fed into an NN to directly compute the output of IT2FLS models. The IT2FNN training is performed using back propagation learning algorithm to minimize the error-based cost function. The optimal weighting factors in the consequent part of this IT2FNN are directly generated from the optimal training algorithm. Weather data are obtained from the Nigeria Meteorological Agency (NMA) and historical electric load data are obtained from the Power Holding Company of Nigeria (PHCN) and used to perform comparative studies. The result of performance of IT2FNN is compared with IT2FLS and IT1FLS methods for short term load forecasting with MSE of 0.00123, 0.00185 and 0.00247 respectively. Also, the results of forecasting are compared using RMSE of 0.035, 0.043 and 0.035 respectively, indicating a best accurate forecasting with IT2FNN. In addition, the result of performance of IT2FNN is compared with IT2FLS and IT1FLS methods for short term load forecasting with MAPE of 1.5%, 3% and 4.5% respectively. It can be concluded that IT2FNN offers encouraging and acceptable degree of accuracy than IT2FLS and IT1FLS in handling the STELF task. This is due to the ability of IT2FNN to cope with uncertainties adequately and is able to tune the parameters of IT2FL in STEL forecasting to give a better solution then. In the future, full implementation of the system can be carried out and the proposed system can be improved by integrating IT2FLS with other algorithms like particle swarm optimization, to handle more uncertainties in STEL forecasting.

## REFERENCES


[1] S. A. Soliman, and A. M. Al- Kandari, “Electrical Load Forecasting Modeling and Model Construction,” Elsevier Inc, 2010.

[2] Ranaweera ‹D. K., Farmer, R.G., and Karady, G.G." Economic impact analysis of load forecasting,"IEEE Trans. on Power Systems, August 1997, Vol. 12, No. 3, pp. 1388-1392.

[3] H. M. Al-Hamadi and S. A. Soliman, “Short-term electric load forecasting based on kalman filtering algorithm with moving window weather and load model,” Electr. Power Syst. Res., vol. 68, no. 1, pp. 47–59, Jan. 2004.

[4] S. J. Huang and K. R. Shih, “Short-term load forecasting via ARMA model identification including non-Gaussian process considerations,” IEEE Trans. Power Syst., vol. 18, no. 2, pp. 673–679, May 2003.

[5] J. D. Cryer, Time Series Analysis. Belmont, CA, USA: Duxbury,1986.

[6] D. Niu, Y.Wang, and D. D.Wu, “Power load forecasting using support vector machine and ant colony optimization,” Expert Systems With Applications, vol. 37, no. 3, pp. 2531–2539, 2010.

[7] A. Khosravi, S. Nahavandi, and D. Creighton, “Load forecasting and neural networks: A prediction interval-based perspective,” in Proc.Computational Intell. Power Eng., B. K. Panigrahi, Ed. et al., 2010, pp. 131–150.

[8] H. Quan, D. Srinivasan, and A. Khosravi, “Short-term load and wind power forecasting using neural network-based prediction intervals,”IEEE Trans. Neural Netw. Learning Syst., vol. PP, no. 99, DOI: 10.1109/TFNNL.2013.2276053.

[9] A. Khosravi, S. Nahavandi, and D. Creighton,“Prediction intervals for short-term wind farm power generation forecasts,” IEEE Trans. Sustain. Energy, vol. 4, no. 3, pp. 602–610, Sep. 2013.

[10] A. Khosravi and S. Nahavandi, “Combined nonparametric prediction intervals for wind power generation,” IEEE Trans. Sustain. Energy, vol. 4, no. 4, pp. 849–856, Dec. 2013.

[11] H. Alfares and M. Nazeeruddin, “Electric load forecasting: literature survey and classification of methods,” International Journal of Systems Science, vol. 33, no. 1, pp. 23–34, 2002.

[12] V. Hinojosa and A. Hoese, “Short-term load forecasting using fuzzy inductive reasoning and evolutionary algorithms,” IEEE Transactions on Power Systems, vol. 25, no. 1, pp. 565–574, 2010.

[13] A. Asar, S. R. Hassnain, and A. U. Khattack, “A multiagent approach to short term load forecasting problem,” International Journal of Intelligent Control and Systems, vol. 10, pp. 52–59, 2005.

[14] L. A. Zadeh “Fuzzy Sets”. Information and Control, vol. 8, pp 338–353, 1965.

[15] O.Castillo and P. Melin, “Recent Advances in Interval Type-2 Fuzzy Systems, vol. 1. USA: Springer, 2012.

[16] A. I. Dharma, Robandi, M H. Purnomo “Application of Short Term Load Forecasting on Special Days using Interval Type-2 Fuzzy Inference Systems: Study Case In Bali Indonesia”. Journal of Theoretical and Applied Information Technology, vol. 49 No.2 20th March 2013. Vol. 49 No.2, 514-424.

[17] A. Tale, A. S. Gusain, J. Baguli, R. Sheikh, A. Badar “ Study of Load Forecasting Techniques using Fuzzy Logic”. International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering. Vol. 6, Issue 2, 512-561, 2017.

[18] H. P. Oak and S. J. Honade “A Survey on Short Term Load Forecasting”. National Conference on “Advanced Technologies in Computing and Networking"-ATCON-2015 Special Issue of International Journal of Electronics, Communication & Soft Computing Science and Engineering. 440-441, 2015.

[19] N. Ebrahimpour and Parsa, A. “Forecasting short-term electricity load by using ANFIS and training it based upon chaotic particle swarm optimization”. Academie Royale Des Sciences D Outre-Mer Bulletin Des Seances Vol. 4 No. 4, pp. 135-141, 2015.

[20] J. Zhao and L. Jiang “Forecasting of Type-2 Fuzzy Electric Power System Based on Phase Space Reconstruction Model”. International Journal of Grid Distribution Computing Vol. 8, No.5, pp.263-272, 2015.

[21] S. I. Oko, Ewim, C. U., Ozeden, H. O., Azuatalam, D. T. “Electricity Load Forecasting in Imo State Using Soft Computing Techniques”. International Journal of Scientific & Engineering Research, Volume 7, Issue 6, 16-23, 2016.

[22] N. Roy and P. Mitra “Electric Load Forecasting: An Interval Type-II Fuzzy Inference System Based Approach”. International Journal of Computer Science and Information Technologies (IJCSIT), Vol. 7 (6), 2515-2522, 2016.

[23] A. Khosravi, S. Nahavandi, D. Creighton, and D. Srinivasan, “Interval type-2 fuzzy logic systems for load forecasting: A comparative study,”IEEE Trans. Power Syst., vol. 27, no. 3, pp. 1274–1282, Aug. 2012.

[24] J. M. Mendel, Uncertain Rule-Based Fuzzy Logic Systems: Introduction and New Directions. Englewood Cliffs, NJ, USA: Prentice-Hall,2001.

[25] H. Hagras and C. Wagner, “Towards the wide spread use of type-2 fuzzy logic systems in real world applications,” IEEE Computational Intell. Mag., vol. 7, no. 3, pp. 14–24, 2012

[26] R. Abiyev and O. Kaynak, “Type 2 fuzzy neural structure for identification and control of time-varying plants,” IEEE Trans. Ind. Electron., vol. 57, no. 12, pp. 4147–4159, Dec. 2010.

[27] Y.-Y. Lin, J.-Y. Chang, N. Pal, and C.-T. Lin, “A mutually recurrent interval type-2 neural fuzzy system (mrit2nfs) with self-evolving structure and parameters,” IEEE Trans. Fuzzy Syst., vol. 21, no. 3, pp.492–509, 2013.

[28] Khosravi, A. and Nahavandi, S. ( 2014). Load Forecasting Using Interval Type-2 Fuzzy Logic Systems: Optimal Type Reduction. IEEE Transactions on Industrial Informatics, Vol. 10, No. 2, May 2014, 1055-1063.

[29] X. Jianmin, S. Yanfang, L. Hongjie (2008). Type-2 Fuzzy Neural Network. Proceedings of the 27th Chinese Control Conference July 16-18, 2008, Kunming,YuFNNan, China, 406-410.

[30] J. M. Mendel, “On the importance of interval sets in type-2 fuzzy logic systems”, Proceedings of Joint 9th IFSA Word Congress and 20th NAFIPS International Conference, Vancouver, BC, Canada, (2011), pp. 1646-1652.

[31] Q. Liang and J. M. Mendel, “Interval type-2 fuzzy logic systems: theory and design”, IEEE Transactions on Fuzzy systems, vol. 8, no. 5, (2012), pp. 534-550.

[32] S.Hemachandra, S. Vidyanikethan, A. Rangampet, Neuro-fuzzy control in load forecasting of power sector, International Journal of Power System Operation and Energy Management ISSN (PRINT): 2231 – 4407, Volume-2, Issue-1,2, 2012.

[33] Yang-Yin Lin, Shih-Hui Liao, Jyh-Yeong Chang, Member, IEEE, and Chin-Teng Lin, Fellow, IEEE (2014)Simplified Interval Type-2 Fuzzy Neural Networks. IEEE Transactions on Neural Networks and Learning Systems, Vol. 25, No. 5, May 2014.

[34] J. Mendel, H. Hagras and R. John Standard Background Material About Interval Type-2 Fuzzy Logic Systems. google scholar, 2006.

[35] J. Mendel and R. John, “Type-2 fuzzy sets made simple,” IEEE Transactions on Fuzzy Systems, vol. 10, no. 2, pp. 117–127, 2002.

[36] Liang, Q. and Mendel, J. M. (2000). Interval type-2 fuzzy logic systems: theory and design. IEEE Transactions on Fuzzy Systems, 8(5), 535–550.

[37] Wu, Dongrui (2005), Design and analysis of Type-2 Fuzzy Logic Systems. A Master’s Thesis, Department Of Electrical Aad Computer Engineering, National University of Singapore 2005.

[38] D. R.Wu and W. W. Tan “Computationally efficient type-reduction strategies for a type-2 fuzzy logic controller”, in FUZZ-IEEE, Reno, USA pp. 353-358, , May, 2005.

[39] N. N. Karnik and J. M. Mendel, “Centroid of a type-2 fuzzy set,” Information Sciences, vol. 132, pp. 195-220, 2001.

[40] A. Khosravi, and S. Nahavandi, Load Forecasting Using Interval Type-2 Fuzzy Logic Systems: Optimal Type Reduction. IEEE Transactions on Industrial Informatics, Vol. 10, No. 2, May 2014 1055-1063.

[41] N. Karnik, J. Mendel, and Q. Liang, “Type-2 fuzzy logic systems,”IEEE Trans. Fuzzy Syst., vol. 7, no. 6, pp. 643–658, Dec. 1999.

[42] H. Wu and J. Mendel, “Uncertainty bounds and their use in the design of interval type-2 fuzzy logic systems,” IEEE Trans. Fuzzy Syst., vol. 10, no. 5, pp. 622–639, Oct. 2002.

[43] D. Wu and J. Mendel, “Enhanced karnik-mendel algorithms,” IEEETrans. Fuzzy Syst., vol. 17, no. 4, pp. 923–934, Aug. 2009.

[44] D. Wu, “Approaches for reducing the computational cost of interval type-2 fuzzy logic controllers: Overview and comparison,” IEEE Trans. Fuzzy Syst., vol. 21, no. 1, pp. 80–90, 2013.

[45] D.Wu and M. Nie, “Comparison and practical implementation of typereduction algorithms for type-2 fuzzy sets and systems,” in Proc. IEEE Int. Conf. Fuzzy Syst., 2011, pp. 2131–2138.

[46] M. Melgarejo, “A fast recursive method to compute the generalized centroid of an interval type-2 fuzzy set,” in Proc. AFNNu. Meeting North Amer. Fuzzy Inf. Process. Soc., 2007, pp. 190–194.

[47] C. Li, J. Yi, and D. Zhao, “A novel type-reduction method for interval type-2 fuzzy logic systems,” in Proc. 5th Int. Conf. Fuzzy Syst. Knowledge Discovery, Jinan, China, 2008, pp. 157–161.

[48] J. Mendel and F. Liu, “Super-exponential convergence of the Karnik–Mendel algorithms for computing the centroid of an interval type-2 fuzzy set,” IEEE Trans. Fuzzy Syst., vol. 15, no. 2, pp. 309–320, Apr. 2007.

[49] K. Metaxiotis, A. Kagiafnnas, D. Askounis, and J. Psarras, “Artificial intelligence in short term electric load forecasting: A state-of-the-art survey for the researcher,” Energy Convers. Manag., vol. 44, no. 9, pp.1525–1534, Jun. 2003.

[50] M. Biglarbegian,W.Melek, and J. Mendel, “On the stability of interval type-2 TSK fuzzy logic control systems,” IEEE Trans. Syst., Man, Cybern. B, Cybern., vol. 40, no. 3, pp. 798–818, Sep. 2010.