SHORT TERM LOAD FORECASTING BASED ON HYBRID ARTIFICIAL NEURAL NETWORKS AND PARTICLE SWARM OPTIMISATION

Name: Ellen Shezi

Supervisor: Prof K.A Folly

Thesis presented for the degree of Master of Science in Electrical Engineering in the department of Electrical Engineering, University of Cape Town

16 February 2015

Declaration

This dissertation is submitted to the Department of Electrical Engineering, University of Cape Town, in complete fulfilment of the requirements for the degree of Master of Science. It has not been submitted before for any degree or examination at this or any other university. The author confirms that this thesis is based on her own work. Portions of this work have been published at refereed international conferences. The author confirms that she was the primary researcher in all instances where work described in this thesis was published under joint authorship.

"I know the meaning of plagiarism and declare that all the work in the document, save for that which is properly acknowledged, is my own".

Acknowledgements

I would like to thank God for His unfailing love and mercies throughout this research journey that I undertook. It was filled with many challenges however I was able to conquer them through Christ who gave me strength.

I would like to thank my husband and children for supporting me during my study periods; I could never have done this without them. I would also like to thank my supervisor Prof KA Folly for the knowledge and information that he willingly shared with me. His guidance has been very beneficial throughout this journey.

I would also like to thank the South African Weather Bureau Services (SAWBS) for providing me with weather data which was necessary for the purpose of this research.

This dissertation is dedicated to my father, SB Banda, who passed away 13 September 2013. He was very supportive of my further studies and always had a word of encouragement for me at all times.

Synopsis

Short term load forecasting (STLF) is the prediction of electrical load for a period that ranges from the next minute to a week. The main objectives of the STLF function are to predict future load for the generation scheduling at power stations; assessment of the security of the power system as well as for timely dispatching of electrical power. STLF is primarily required to determine the most economic manner in which an electrical utility can schedule generation resources without compromising on the reliability requirements, operational constraints, policies and physical environmental and equipment limitations. Another application of the STLF is for predictive assessment of the power system security. This system load forecast is an essential data requirement for off-line network analysis in order to determine conditions under which a system may become vulnerable. This information allows the dispatcher to prepare the necessary corrective actions. The third application of STLF is to provide the system dispatcher with more recent information i.e., the most recent forecast with the latest weather prediction and random behaviour taken into account. The dispatcher needs this information to operate the system economically and reliably.

Due to the sensitivities surrounding a load forecast, it thus becomes crucial that the forecasting error is minimised. There are various methods that are used for short term load forecasting, namely; statistical methods and computational intelligence methods. Statistical methods are known as the regression methods which forecast the future electrical load based on historic time series load information. These methods have been in use for many years however due to the dynamic changes in the power system today such as the introduction of Independent Power Producers (IPPs) onto the grid; it becomes difficult to use these methods because they are very static and inflexible i.e. they cannot be manipulated by including rules or expert knowledge in order to counter the effect of any sudden changes in the power system. Their inability to adapt to the changing behaviour of the power system thus leads to high forecasting errors.

Computational intelligence (CI) methods however are dynamic and are able to learn by experience. Short term load forecasts have been conducted by using various CI methods such as Artificial Neural Networks (ANNs), Genetic Algorithms (GAs), Fuzzy Logic (FL), Expert Systems (ES), and Particle Swarm Optimisation (PSO). Hybrid versions of these methods, where two or more CI methods are amalgamated in a process to forecast future load, have also been used.

In this research, a traditional forecasting technique, Multiple Linear Regression (MLR), was compared with a CI technique, Artificial Neural Networks. ANN was also compared with another neural network method namely Elman Recurrent Neural Network (ERNN) to determine whether a more neural network method with memory yields better results as compared to ANN. The inputs of each of the forecasting models consisted of load affecting weather variables (humidity and temperature). A correlation study was conducted between these variables and the load in order to determine the strength of the relationship. Two forecasters (weekend and weekday) with three variations of inputs, namely; load only (ANN, ERNN), load plus temperature (ANN-t, ERNN-t where t represents temperature) and load plus temperature and humidity (ANN-w, ERNN-w where w represents temperature and humidity), were tested and results were compared.

These results were also then compared with an MLR method. The performance evaluation of the forecasting methods was conducted by using mean absolute percentage error (MAPE). ANN results showed a minimum forecasting error of 4.32% and a maximum forecasting error of 10.96% while ANN-w load predictions produced minimum and maximum values of 3.55% and 10.33% respectively. MLR produced fairly good forecasts with a minimum error of 3.1% and a maximum of 13.3%. ERNN, ERNN-t and ERNN-w produced minimum and forecasting errors of 6% and 14.1%, 5.15% and 13.12%, 5.52% and 14.17% respectively. It was found that the best performing method out of all the tested techniques was the temperature sensitive ANN as it was able to forecast with a minimum forecasting error of 3.05%, a maximum forecasting error of 9.16% and an average error of 5.5%. The performance of this ANN further proved that temperature plays a major role as a load affecting variable.

One of the objectives of this research was to obtain load forecasts with a forecasting error of ±5%. A hybrid method, consisting of the temperature sensitive ANN and PSO, was thus tested in an attempt to improve the forecasting error. PSO was utilised to alter the weights of the ANN such that the resulting mean square error for the training data is reduced. The performance of the hybrid method was found to be a greater improvement as compared to the ANN-t performance with a minimum forecasting error of 2.51% and a maximum of 5.70%. The average forecasting error for PSO-ANN was 3.89%. This reduction in forecasting errors proves that by introducing hybridization of CI techniques, better forecasting results can be achieved.

Table of Contents

Table of Figures

List of Tables

Chapter 1 : Introduction

Load forecasting plays a vital role within an electrical utility and is required for management of energy within a power system. There are three categories of load forecasting namely; short term, medium term and long term load forecasting. A short term load forecasting ranges from minutes up to a week while the medium and long term range from a week to a year and a year up to 20 years, respectively. A short term load forecast is used by system dispatchers and operation analysts to control and plan power system operations. They are also very important for power system security studies such as contingency analysis and load management and are very useful within a power distribution environment.

There is thus a need for accuracy with regards to load forecasts as the degree of accuracy can have significant effects on power system operations as the economy of operation and the control of the power system may be quite sensitive to forecasting errors. Accurate forecasts can ensure that a utility is able to reduce its generation costs by assisting the operators in making accurate decisions regarding the purchasing of energy as well as scheduling equipment maintenance outages. Large forecasting errors can have an adverse effect on the power system; for example if a forecast exceeds the amount of load demand, this may result in the start-up of too many generation units and unnecessarily high levels of reserves while forecasts that are too low, may result in failure to provide the necessary spinning and operating reserves as well as not meeting the load demand.

All these factors can induce heavy economic and operational costs. Short term load forecasting also plays an important role in the reliability of the power system; power system operators use load forecasting as a base to which they can determine whether the system is vulnerable and also helps them to determine any overload conditions and whether more generation is required to meet the customers' demand. They can be able to develop mitigating actions as a result.

The problem of short term load forecasting is still a challenge despite the numerous literatures that are available. There are a number of factors that affect the demand that are either unknown or random; in particular the forecasting of load for special occasion days such as holidays, days on which strikes occur or extreme weather conditions; these are difficult to translate to a mathematical model or use in a typical load forecasting tool. However, with the new developments within the computational intelligence (CI) field, it has become possible to produce improved forecasting results as compared to time-series based forecasting tools. CI tools are able to learn and adapt to changing behaviour of the load and determine the relationships between input and output variables of which the typical time-series based tool had a difficulty.

CI techniques have been used in the past to conduct short term electrical load forecasting. A number of researchers have shown that it is possible to obtain good forecasting results by using CI techniques such as Artificial Neural Networks or hybrid methods which consist of two or more CI methods. This research investigates the application of Artificial Neural Networks (ANN), Elman Recurrent Neural Networks (ERNN) and Particle Swarm Optimization (PSO) to the problem of load forecasting using actual data of an electrical utility in South Africa.

1.1 Objectives of research

The main objective of this research is to develop short term load forecasting models using CI techniques. It is required that the models should incorporate load affecting factors namely weather (temperature, humidity, rainfall).

The objectives are as follows:

- Develop ANN, ERNN and PSO based hybrid models
- Apply actual load data obtained from Eskom Distribution in Kwa-Zulu Natal for a specific substation and predict next day half hourly load profile using the developed models.
- Perform comparative studies between the CI techniques.
- To produce a fairly accurate load forecast with an error of ±5%

1.2 Scope and Limitations

The scope of this thesis is limited to the research of three CI methods namely ANN, ERNN and PSO as applied to short term load forecasting. These methods were chosen based on the accessibility of the toolboxes which are found in MATLAB (the platform that was used to conduct the design, validation and testing of the load forecasting). The major limitation experienced in this research was the lack of complete rainfall data from the South African Weather Bureau Services (SAWBS). This weather variable could not be included in the forecasting models as some information for certain days in a month were missing.

1.3 Thesis outline

Chapter 1: Provides an introduction to the need for short term load forecasting for an electrical utility and a brief discussion on the impact of accurate forecasting. The background and objectives of the thesis are described and form the basis upon which this research was conducted. The research methodology as well as the scope and limitations are presented.

Chapter 2: Presents a literature review on Computational Intelligence techniques that are commonly applied to STLF is presented in this section. It presents the following techniques: Artificial Neural Networks, Fuzzy Logic, Genetic Algorithm, Expert systems and Particle Swarm Optimisation. The literature survey presents the results obtained using these techniques as well as any drawbacks that the authors may have encountered. A discussion on traditional forecasting methods is also undertaken.

Chapter 3: Introduces the application of Artificial Neural Networks (ANN) to the problem of STLF. A detailed background on the network topology and elements that make up an ANN is presented. Two forms of ANN networks; namely, feed forward ANN and Elman recurrent network (ERNN), are used to test the technique. The models used to conduct the analysis are presented and the simulation results from each of the models are described. The two methods are compared to determine the best performing method. ANN is also compared to a Multiple Linear Regression model.

Chapter 4: Introduces the application of Particle Swarm Optimisation (PSO) to STLF. An overview of PSO is presented. In this section, PSO is applied in conjunction with the feed forward ANN network presented in chapter 3 in order to further optimise the forecasting error. These systems are all tested using the same data applied in chapter 3.

Chapter 5: This section presents the conclusions and recommendations for future work.

Chapter 2 : Overview of Short Term Load Forecasting

This section describes the various short term load forecasting (STLF) techniques that are typically used with a focus on computational intelligence (CI). It also describes the factors that influence the shape of the load profile. A survey of literature on the use of CI methods for STLF is presented.

2.1 Background of Short Term Load Forecasting

Short term load forecasting is the prediction of load for a period from minutes to a maximum of a week. It is required for the economic dispatch of generation as well as for power system security studies which include contingency analysis and load management [1, 2, 3]. Since the availability of electricity plays a vital role in the economic development of a country, it is imperative that an electrical utility be able to produce an accurate load forecast in order to meet the power requirements of that country as well as to support its development [2, 4].

In order to develop an accurate forecasting tool, it is essential to understand the characteristics of a power system load. There are various factors that affect the shape of a load profile. A load profile is a curve on a chart which depicts the trend in the supply of electrical power in Watts over a time period [5]. A load profile can be influenced by the following factors:

Time: The power system load behaves differently at various times in a day i.e., a 24 hour period. The load at midnight will be considerably different from the load measured at peak hours (i.e. 18:00pm) of the same day. Figure 2.1 shows a typical load profile for a predominantly residential area and Figure 2.2 depicts the load profile for a small municipality that supplies light commercial, industrial and residential customers. From Figure 2.1, it can be seen that the load peaks between the hours of 5:00am -08:00am (this would correspond to a morning peak where people are getting ready for work and residential geysers are being utilised). There is also a large evening peak which occurs from approximately 17:00pm to 21:00pm (this would be the time people get back home and begin cooking, washing, etc.).

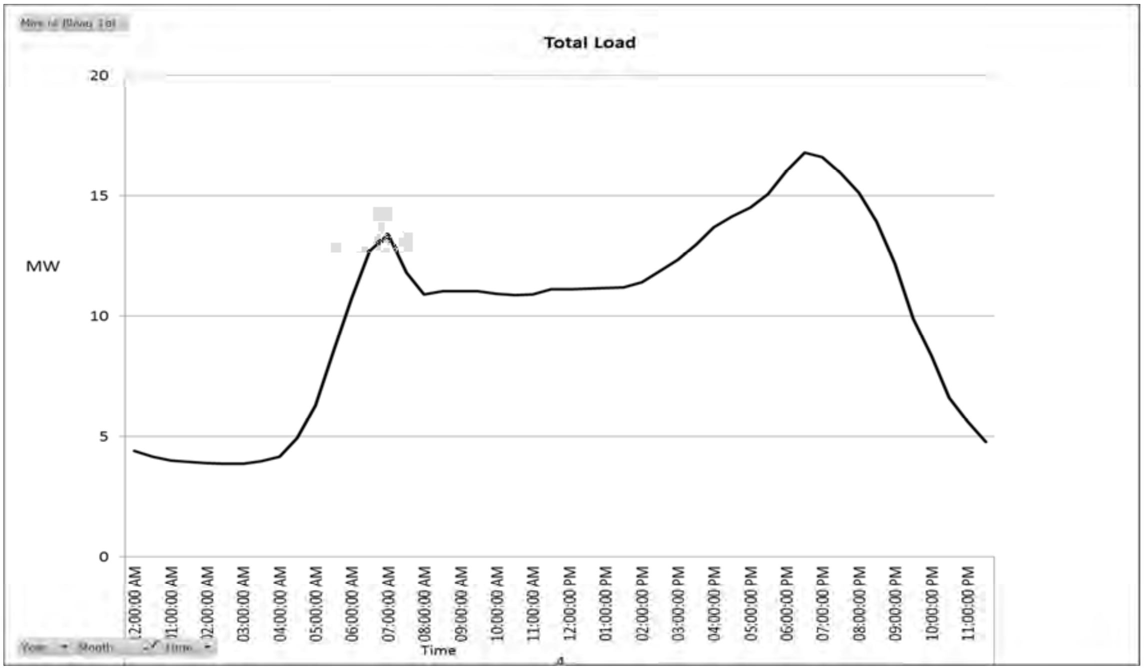


Figure 2.1: Typical residential load profile

Figure 2.2 differs from Figure 2.1 in that the morning and evening peaks are not as clearly defined as in Figure 2.1. This is largely attributed to the customer mix of commercial and industrial load that is supplied. Some of these customers have operations or processes that operate on a 24 hour basis while other operations generally operate between 7:00am and 18:00pm.

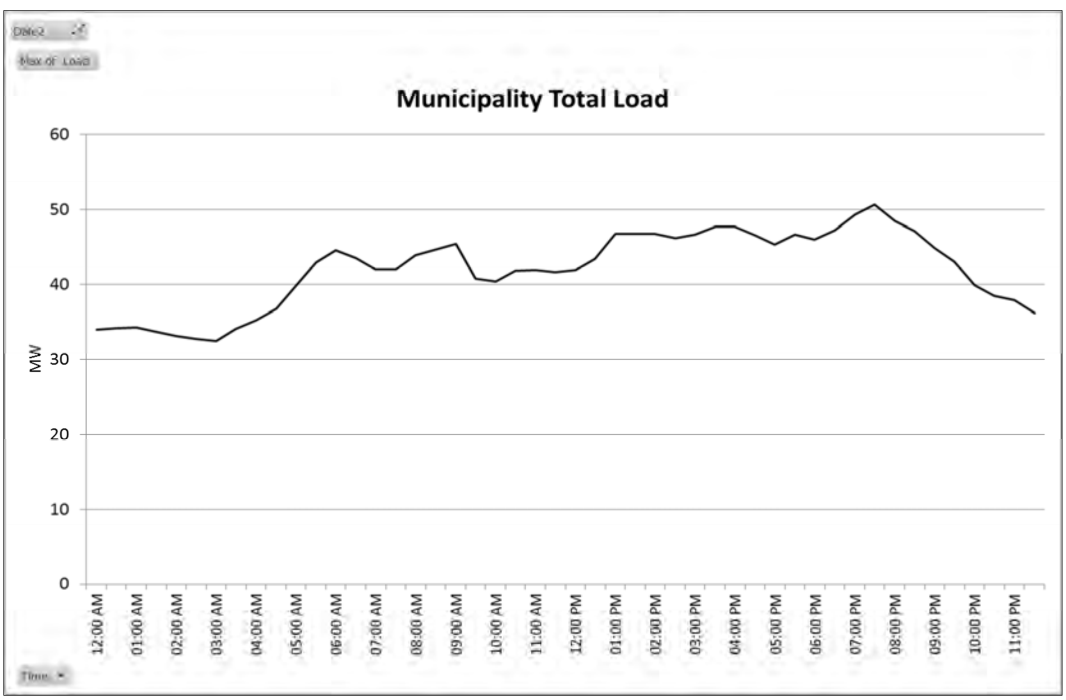


Figure 2.2: Load profile of a small municipality

Day type: This has a relatively high influence on the load behaviour as generally a weekend load is lower than a weekday load. This can be confirmed by Figure 2.3. It can be seen that weekend (i.e. Saturday and Sunday) load profiles are generally much lower than weekday (i.e., Tuesday, Wednesday) load profiles. Weekends are generally reserved for leisure while weekdays are mainly for work where industries are operational. A shift in the time for the morning peak can be seen in Figure 2.3 below, this is evident of a weekend load where people tend to wake up a little later than they would during the working days. Also, during weekend, most industries are closed.

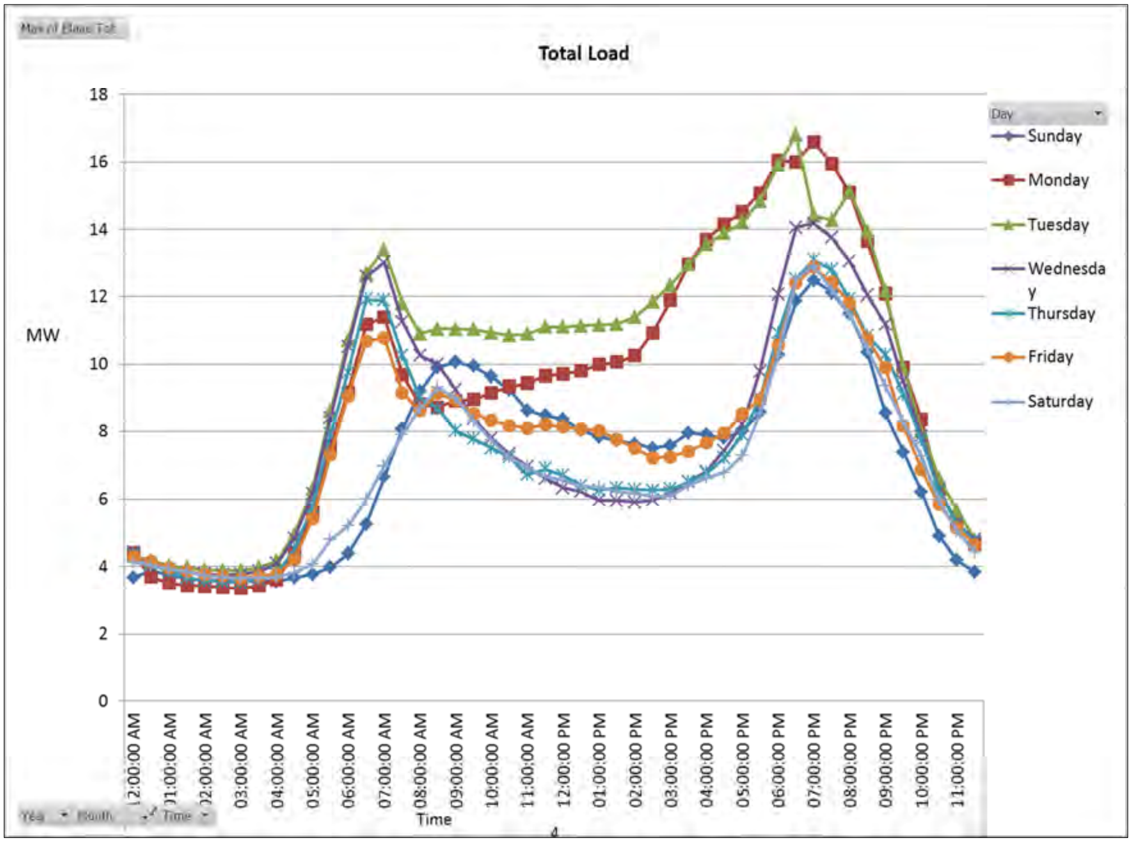


Figure 2.3: Month/Day type load profile

Seasonality: Seasons (i.e., summer vs. inter), play a major role in influencing the consumption behaviour of a customer. In winter, the nights are longer and the sun sets very early leading to lights coming on earlier for a much longer duration as compared to summer when the sun sets late. The demand for power in winter is much higher than in summer due to the drop in temperature. Figure 2.4 depicts a year load profile taken from August 2010 to August 2011; the numbers 1 to 12 correspond to the months of the year i.e. January to December, respectively. It can be seen that the winter load profiles (June, July, August i.e. months 6, 7 and 8) are much higher compared to the summer load profiles (December, January and February i.e. months 12, 1 and 2). This would be as a result of the cold; most customers utilise their heating equipment during the winter season to try and keep warm thereby increasing the load demand while in summer, most utilise natural resources such as opening windows to keep cool on hot days.

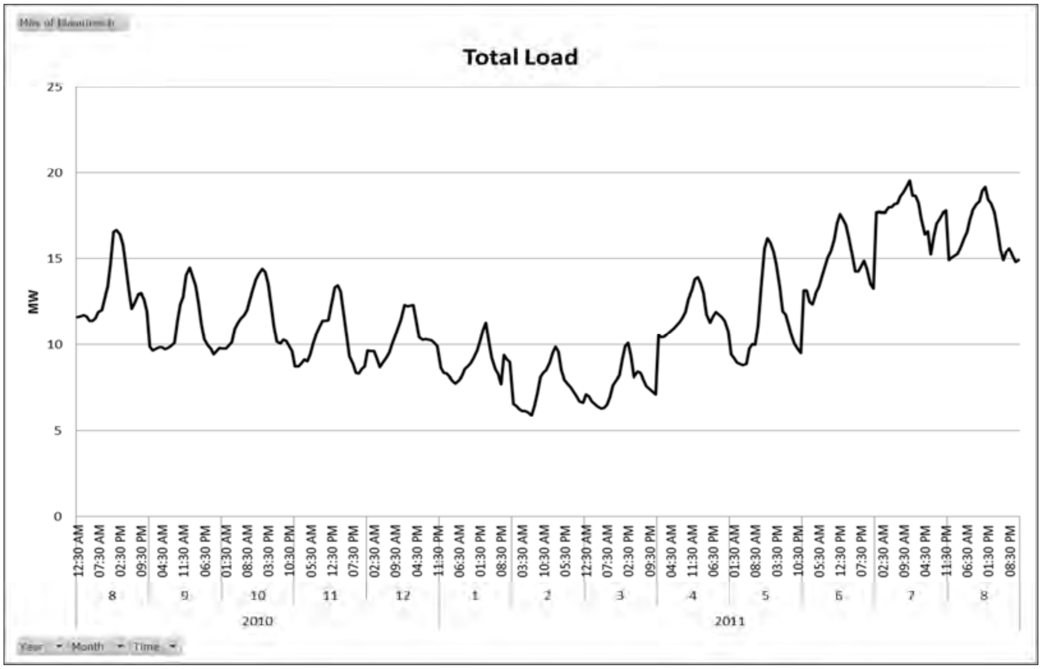


Figure 2.4: Seasonality load profile

Weather: Temperature, humidity, precipitation, wind speed, cloud cover etc., are elements that make up the weather factor which influences the load profile. The change in weather defines how people feel which then translates to the type of appliances they may use to regain a particular sense of comfort such as using heaters when it gets cold or switching on the air conditioner on a hot day.

Economy: The economic situation within a country influences the utilization of electricity. As a country develops, more commercial/industrial companies develop thus increasing the demand as well as when an economy declines so the degree of investment and new business development decreases leading to a decrease in electric load demanded. According to literature, the economy influences a medium and long term load forecasting more than it does on a short term load forecast due to the period that it forecasts. In this research, this factor is not taken into consideration since the forecast is for the next 24 hours.

Electricity Pricing: The price of electricity plays a major role in determining the way a customer uses his/her electricity. The higher the cost of electricity, the more a customer is determined to use alternative energy such as installing a generator or using solar energy as the main power source in the household and use electricity from a utility as a back-up source. The electricity demand is then reduced as a result of the savings initiatives brought about by the customer. On the other hand, if cost of electricity is relatively cheap, one will find more customers applying for electricity connection resulting in an increase in power system demand. Customers can thus be able to adjust their consumption behaviours based on the electricity price.

Public Events: Examples of these are nationwide strikes and sports events. Strikes have the effect of decreasing the load as once a strike is announced; employees tend to stay away from work thereby reducing the load consumption depending on the type of industry affected. Sports events have an impact on the load during the sports fixtures only and have the effect of increasing the load consumption.

Holidays: This variable can be split into two categories namely public and school holidays. School holidays tend to affect the weekday load profile particularly in the mornings as the school children are no longer waking up to get ready for school, there may be a shift in the morning peak as a result. Demand on public holidays is generally much lower than on “normal days”. A model to forecast public holidays may be required since their demand is different from a “normal day”.

The above-mentioned factors can be incorporated into a load forecasting tool based on history and known information however anomalies do occur which could cause deterioration in the accuracy of the forecast.

Some of the anomalies that can occur are listed below:

1. Interruption of supply – there could have been a temporary loss of supply to certain parts of the network
2. Switching events – the network could be reconfigured for planned maintenance or as a result of an outage
3. Demand side management (DSM) initiatives – customers may be implementing certain DSM initiatives within their homes/work places in order to reduce their energy bills

These are a few anomalies that are generally difficult to incorporate into a short term forecasting tool and may cause unwanted variances. Therefore, when developing the forecasting tool, it is important to include a random variable that may attempt to simulate these anomalies. This research utilises the weather as the load affecting factors. A correlation study was conducted in order to determine which of the weather variables plays a major role in shaping the load profile.

2.2 Short Term Load Forecasting Techniques

Short term load forecasting techniques can be classified into two categories, namely Statistical and Computational Intelligence (CI) methods. Statistical methods such as Multiple Linear Regression, Stochastic Time Series, State Space Method, General Exponential Smoothing, [4, 6] have been widely applied to short term load forecasting. Statistical load forecasting tools utilize time series models which extrapolate historical load data to predict the future loads. These tools assume a static load series and retain normal distribution characteristics. Due to their inability to adapt to changing environments and load characteristics, large forecasting errors would result when a deviation between historical load data and present conditions occurs [7]. CI techniques are able to learn and adapt to changing environments and forecast accordingly with less forecasting errors as compared with statistical forecasting tools [3].

Hybrid approaches whereby two or more CI techniques are combined to bring about a load forecasting tool are also avenues that have been explored [8]. For example, Fuzzy Logic can be combined with ANNs [9, 10], PSO with ANN [11, 12, 13, 14], PSO with Fuzzy logic [15], Expert systems [16, 17] or there can be a mixture of GA with ANNs [8, 18], etc. These hybrid methods have not been widely applied to short term load forecasting. However, some literature proposes that the use of two or more CI techniques would be beneficial in reducing forecasting errors [16]. The benefit is in utilizing the advantages of these techniques in such a way as to complement each other with the ultimate goal of obtaining a better forecasting accuracy [16].

2.2.1 Statistical Approaches for STLF

Statistical approaches to STLF can be classified into two categories; time series and regression methods [19]. In time series, the future load is predicted by using time series analysis techniques because the load is treated as a time series signal. Regression based methods look at the shape of the load profile and acknowledges that the load is dependent on weather variables. The future load is then predicted by using weather variables in conjunction with load data into a function that has calculated the relationship between the two pieces of information.

2.2.1.1 Time Series

This idea explains that a load pattern is nothing more than a time-series signal with known seasonal, weekly and daily periodicity, which gives a prediction of the load at the given season, day and hour of the day.

Time-series models consist of three general classes [4, 8, 19, 20]:

1. Auto regressive (AR) model where the current value of the time series output is represented linearly in terms of its previous values and a random noise factor.
2. Moving average (MA) model where the current value of the time series is expressed linearly in terms of the current and previous values of white noise.
3. Auto regressive moving average (ARMA) model where the current value of the time series is expressed linearly in terms of its values at previous periods and in terms of its current and previous values of white noise.

2.2.1.2 Regression Analysis

Regression technique can be defined as the analysis of relationship among variables. The relationship is expressed in the form of an equation connecting the response or dependent variable and one or more independent variables [4]. Most regression approaches try to find the relationships between weather variables and current load demand. The conventional regression approaches use linear or piecewise-linear representations for forecasting functions. By a linear combination of these representations, the regression approach finds the functional relationships between selected weather variables and load demand [6, 19].

2.2.2 Computational Intelligence Techniques

This section discusses the types of CI techniques that have been applied to short term load forecasting and describes the advantages and disadvantages of each of the techniques. Four commonly used techniques were chosen for this discussion namely; Expert Systems, Genetic Algorithm (GA), Artificial Neural Networks (ANN) and Fuzzy Logic (FL) systems.

2.2.2.1 Artificial Neural Networks

Artificial neural networks are a type of computational intelligence which is inspired by the way the biological systems of humans such as the brain, process information. The human brain is made up of neurons which are interconnected by dendrites and collects information via this connection. ANN's are made up of a number of simple and highly interconnected processing elements called neurons [21]. An illustration of a neuron is shown in Fig. 2.5. All the neurons in the brain work in unison to make sure that all the information that is received is processed as efficiently and accurately as possible [21, 22]. ANNs learn by example and are configured for particular classes of problems or applications through a learning system [22].

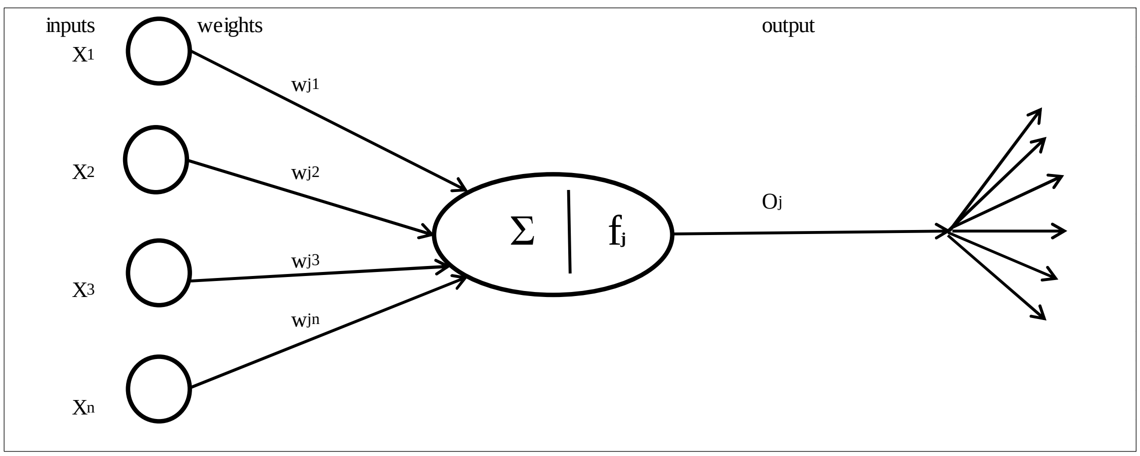


Figure 2.5: Structure of neuron [21]

From Figure 2.5 it can be seen that a neuron accepts inputs ($x_1 - x_n$). These inputs are connected to the neuron via the weights ($w_{j1} - w_{jn}$). The weights depict the strength of the connection between the input variables and the output. The inputs together with the adjustable weights are then summated and then taken through a linear or non-linear transfer function (f_j) giving rise to an output (o_j).

The equation for the output is as follows:

$$o_j = f_j \left(\sum_1^n (x_n * w_{jn})\right) \quad (2.1)$$

There are a number of topologies that are utilized by the ANN such as feed-forward network. Feed-forward is basically the structure where the input signals is propagated in one direction i.e. from input neuron via a hidden layer to the output layer. Figure 2.6 illustrates a feed-forward neural network which is the most commonly applied neural network architecture to short term load forecasting. It comprises of an input vector which would generally contain your inputs made up of historical load data, historical and forecasted weather parameters, day types as well as other load affecting factors [23]. It contains a hidden layer (you can have more than one hidden layer) and then an output layer, usually one output is sufficient; however this can be configured as required and is generally problem specific.

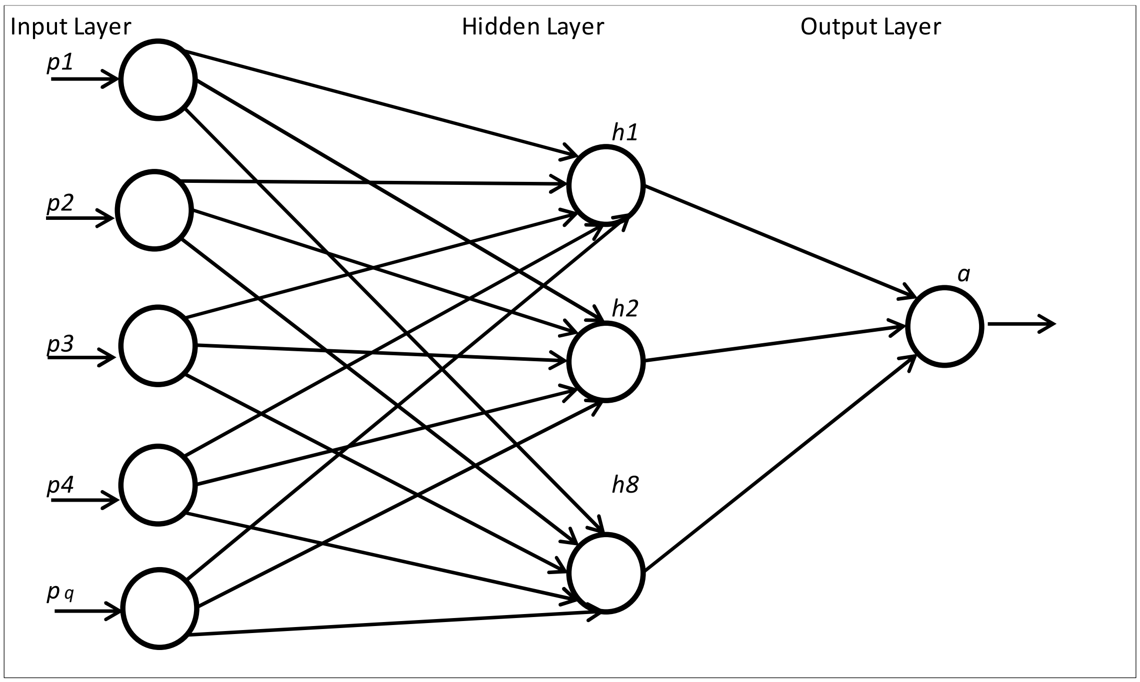


Figure 2.6: Feedforward Network Topology [23]

A feedback ANN also known as Recurrent Neural Network (RNN) also exists where the networks have signals traveling in both directions within the network [24]. An example of these networks is the Elman Recurrent Network [25]. Feedback networks are dynamic and their status is always changing until an equilibrium point is reached. They can become very complicated but are regarded as very powerful networks [24, 25]. Figure 2.7 illustrates the topology of a recurrent neural network.

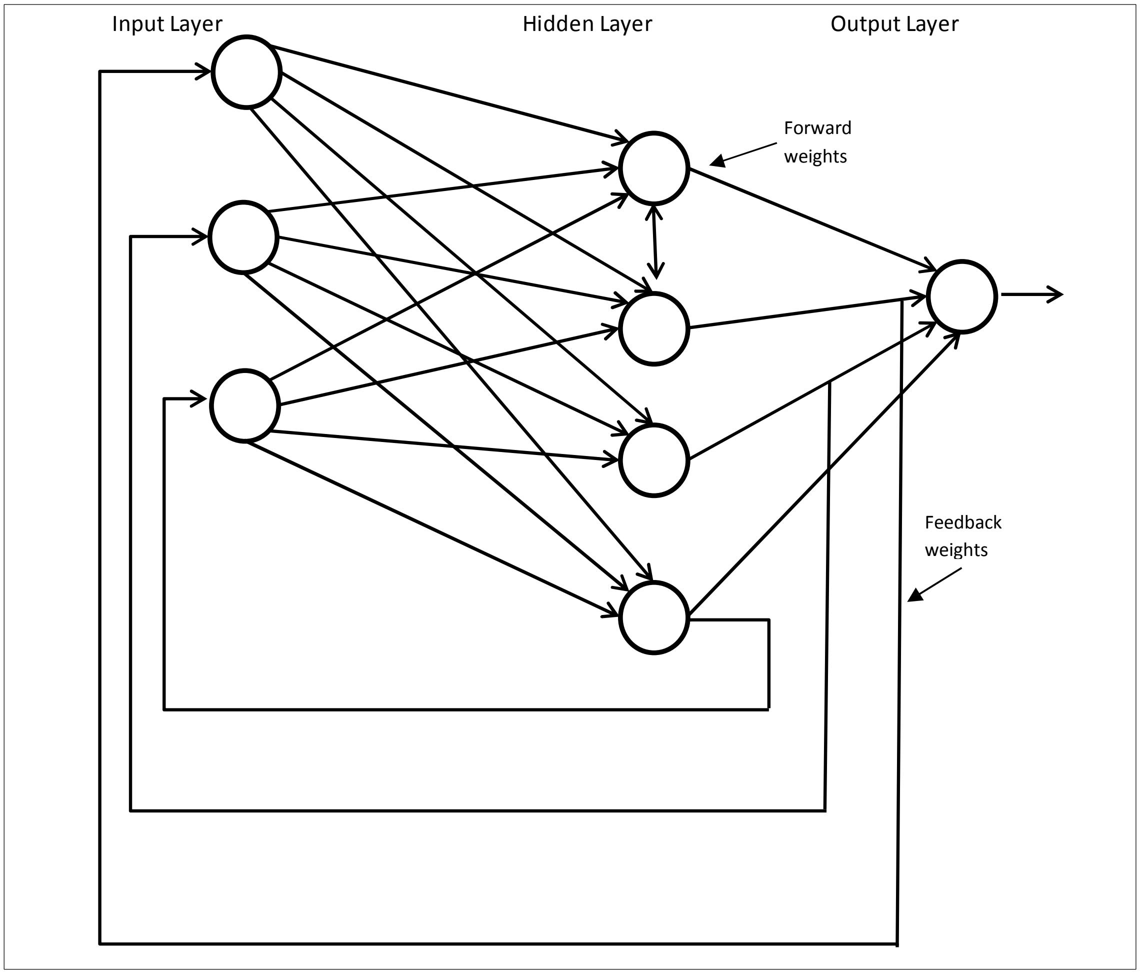


Figure 2.7: Recurrent Neural Network topology [25]

A training algorithm such as the back propagation is used for ANNs. It is basically a training method which finds the difference between what the output is and what it was supposed to be, i.e. the reference or target output. The weights are then adjusted according to the errors and then propagated back into the system until the error is minimised [21].

The back propagation algorithm is excellent in its ability to accommodate weather variables and other variables as deemed fit by the engineer. However, the main drawback with this training algorithm is that the training process can become very cumbersome and time consuming. Convergence of the system also becomes a cause of concern with this algorithm as discussed in [26, 27]. Different methods of curbing the convergence problem are presented in [26] such as the back propagation algorithm with a momentum factor. It is proposed that the inclusion of a momentum factor causes the neural network to converge much faster and introduces a new modified total error function within the algorithm. Since back propagation is based on the gradient descent which is local search algorithm, the solution obtained with this method may find the local minimum and not the global minimum. Training methods based on Evolutionary Algorithms such as GA, PSO have been proposed [18, 28] to solve the problem of the BP algorithm settling into a local minimum.

The most important aspect in creating an accurate neural network is in the selection of input variables. Kandil *et al.* [20] recommends applying statistical analysis to determine which variables have a significant influence on the system load. Zhang *et al.* [23] grouped the input load data into weekday, festival and weekend in order to obtain a greater level of accuracy. Conventional ANN models utilize forecasted weather input variables to predict future load, while this is the most common practice, it can lead to large forecasting errors in case there is a large change in temperature. Osman *et al.* [29] proposes correlation of weather data with load data to determine the input parameters of the network much the same as Kandil *et al*.

Another challenge with ANN models lays with the development of the network topology i.e. the number of hidden layers and neurons. These have a great effect on the learning capability of a neural network and the size of the network may also be dependent on the system that the model would be applied [29]. Therefore, it becomes important to select the network topology very carefully in order to ensure that the quality of outputs is not hampered.

2.2.2.2 Fuzzy Logic

Fuzzy logic is a computer framework based on rules and set theories and reasoning. It implements human experiences and preferences through fuzzy rules embedded in a system. It is a problem solving methodology that helps to come up with conclusions based on ambiguous, imprecise or “noisy” data. The structure of fuzzy inference consists of three conceptual components, namely [30]:

- **Rule Base**: containing a selection of fuzzy rules e.g. ‘if then statements’
- **Database**: defines the membership functions.
- **Reasoning mechanism:** performs the inference procedure upon the rules and given facts and derives a reasonable output or conclusion.

A defuzzification stage is implemented in order to obtain a “clean” output in order to retrieve an unambiguous solution.

Fuzzy Logic has been applied to short term load forecasting by many authors [9, 16, 17, 27]. In [9] Rothe *et al.* proposed a hybrid model for short term load forecasting consisting of a Fuzzy Logic system, Genetic Algorithms and ANN systems. Their research produced results with minimal forecasting errors in the range of 1 to 3%.

In [10] Barkitzis *et al.* developed a fuzzy logic system with a network structure and training procedure like that of a neural network thus giving it the name Fuzzy Neural Network (FNN). The authors discovered that the FNN forecasted future loads with an accuracy that was comparable to that of a neural network. The error ranged from 2.43% to 3.06% for the FNN while their ANN error ranged between 2.3% and 3.14%. Dash *et al.* [16] presents a self-organizing fuzzy neural network which combines the NN's ability to learn and organize data as well as the reasoning capabilities of the Fuzzy logic system. The results of which also proved successful with a very low forecasting error, i.e., less than 2%.

One of the advantages of the fuzzy logic model is that there is no need to have a mathematical formula to map the inputs to the outputs and the fact that imprecise data can be used. However, a very important factor in the fuzzy logic system is the mapping of similarities between certain variables. The disadvantage is that it can prove to be quite problematic when data pairs are not available or little information is used to extract similarities as pointed out in [27]. This is still a problem that needs to be solved for the fuzzy logic system to be implemented successfully.

Various combination approaches incorporating Fuzzy Logic and other CI approaches have also been developed as described previously. These are commonly known as hybrid techniques. Yuill *et al*. [31] discusses one such a model called the Adaptive Neuro Fuzzy Inference System (ANFIS). ANFIS is a multilayer feed-forward network built with fuzzy expert rules provided by a power system expert. This technique has proven to be quite effective in providing accurate results as discussed in [31]. The authors also detail further work that could be applied to the ANFIS hybrid model such as more weather related input variables as well as creating separate ANFIS models for different day types or load types i.e., domestic and industrial loads.

Another hybrid model presented by Kim *et al.* [17] comprises of a fuzzy expert system and an artificial neural network. Two procedures are described in [17], where, in order to obtain an output, the data first passes through the ANN and thereafter this output then goes through the fuzzy system.

The outputs of the fuzzy system are modified based on the possibility of load variation due to changes in temperature and load behaviour on a holiday or special occasion day.

The advantages of hybrid models, in particular those incorporating Fuzzy logic, is that the rules that are built into the forecasting model can be changed when needed. This could also be viewed as a disadvantage if the rules are set by a human whose knowledge may possibly fall short but on the other hand, as newer experiences with the power system are obtained, these can be easily incorporated into the rules. Since ANNs are generally used to train the data, the chances of obtaining high forecasting errors with hybrid techniques are minimal. More fuzzy rules can be added into the model in order to improve accuracy.

2.2.2.3 Expert Systems

Expert systems are techniques that have emerged as a result of advances in the field of computational intelligence in the last three decades. An expert system is a computer program which has the ability to reason, explain, and have its knowledge base expanded as new information becomes available to it much like a human expert [4]. This technique has been implemented for short term load forecasting in [4]. The Expert system based model was created to forecast hourly load data by selecting a reference day load curve according to a set of pre-defined rules. This reference day is then re-shaped according to other sets of rules in order to account for the difference in weather between the reference and forecast day. The authors in [4] concluded that there is potential in the application of expert system to short term load forecasting. Improvements can be brought about by increasing the knowledge of the system and inserting this information as rules into the model.

The load forecasting model is built using the knowledge about the load forecasting domain from an expert in the field. The basis of this technique lies with the knowledge and experience obtained from system operators who would then be called "knowledge experts" [32]. An expert system consists of the following: inference engine, user interface and knowledge base [33].

This knowledge is represented as facts and rules which are then built in the knowledge base component of the expert system. The search for a solution or reasoning about the conclusion drawn by the expert system is performed by the "Inference Engine" component. Each expert system is required to have the capability to trace its reasoning if asked by the user. This facility is built through an explanatory user interface component [4, 33]. Like in the case of Fuzzy logic, expert system can also be combined with ANN. McDonald *et al.* [34] tested the application of ANNs to STLF. The author's observation of the pure ANN models that were constructed and tested was that there was room for improved forecasting which lead to the introduction of an expert system. The resulting forecasting errors ranged from 0.5% to 2.5%.

Liang *et al.* [35] presented a knowledge based expert system for short term load forecasting. Observation of load demand and power system operator (expert) knowledge of the system over a period of time is used to create and establish day types that the expert system would be able to use. The results presented in the paper illustrate that the load forecasting using this method was very accurate as compared to the traditional forecasting technique.

Expert systems make it possible to incorporate knowledge into the algorithmic forecasting models used in the operation of electrical energy systems, and also allow new rules to be included as changes occur. In addition, when queried, the expert system is able to explain and provide reasons. The drawback is that extensive knowledge of the power system is required in order to accurately depict the behaviour of the network as well as to forecast future loads; and the expert system relies heavily on the human expert to arrive at conditional rules [8].

2.2.2.4 Genetic Algorithms

Genetic Algorithms (GAs) are based on the principle of survival of the fittest and is a CI method which simulates the process of organic evolution [36]. It is a computerized search and optimization algorithm based on the principal of natural selection. It operates on a population of individuals which are coded variables representing potential solutions to a given problem [18]. GAs search for solutions in several regions thereby increasing the probability of global convergence [18] as compared to an ANN trained with back propagation algorithm which often suffers from the problem of settling on local minima.

Genetic operators such as crossover and mutation are then applied to the populations to evolve to new possible solutions. There are variants of crossover, one such example is a single point crossover where a point is selected in the parent chromosome to represent the cross over point and all data beyond that point is swapped between the two parent chromosomes. Another form of crossover is called a two point crossover; this is when two points are selected and a binary string from the beginning of chromosome to the first crossover point is copied from one parent, the part from the first to the second crossover point is copied from the second parent and the rest is copied from the first parent [36]. The results of the crossovers are the children. In mutation, GA randomly changes some of the genes values of the parents [7]. Figure 2.8 illustrates the GA algorithm process.

Wu *et al*. [28] presents a short term load forecasting model based on the Elman Recurrent Neural Network (ERNN) with a Genetic Algorithm (GA) being used to train the neural network weights. The authors described the results of this technique as very successful as they were able to attain higher forecasting precision. The only short coming with the research undertaken in [28] is that it only takes into account historical load data. It does not consider any weather variables as well as any other load affecting variables.

Huo *et al.* in [38] presents a load forecasting method based on GA. The authors point out that it is necessary to clean the historical data that would be used in GA in order to obtain an accurate forecast. This is done by removing any bad data that could exist as a result of a fault on the data channel or a remote system fault. This data correction was done by means of two methods which incorporated the replacement of bad data with that of the last period for the same instance as well as the use of a filter which removed random errors. The same can be applied to all load forecasting techniques as the quality of the input data has a high influence on the output of the models.

Yang *et al.* in [39] developed a Genetic Algorithm with neural network and considered the influence of climate on the load forecasting as one of the input variables. Considerable precision was obtained by using this hybrid technique. The weights of the neural network were trained using GA until the learning error stabilized to a certain value and then the back propagation method was applied in order to complete the forecasting process. The idea of utilizing the GA in order to train the network was because the GA searches globally for a solution and would find the best fit weights for the problem as compared to back propagation methods which generally have a problem of settling into a local minimum.

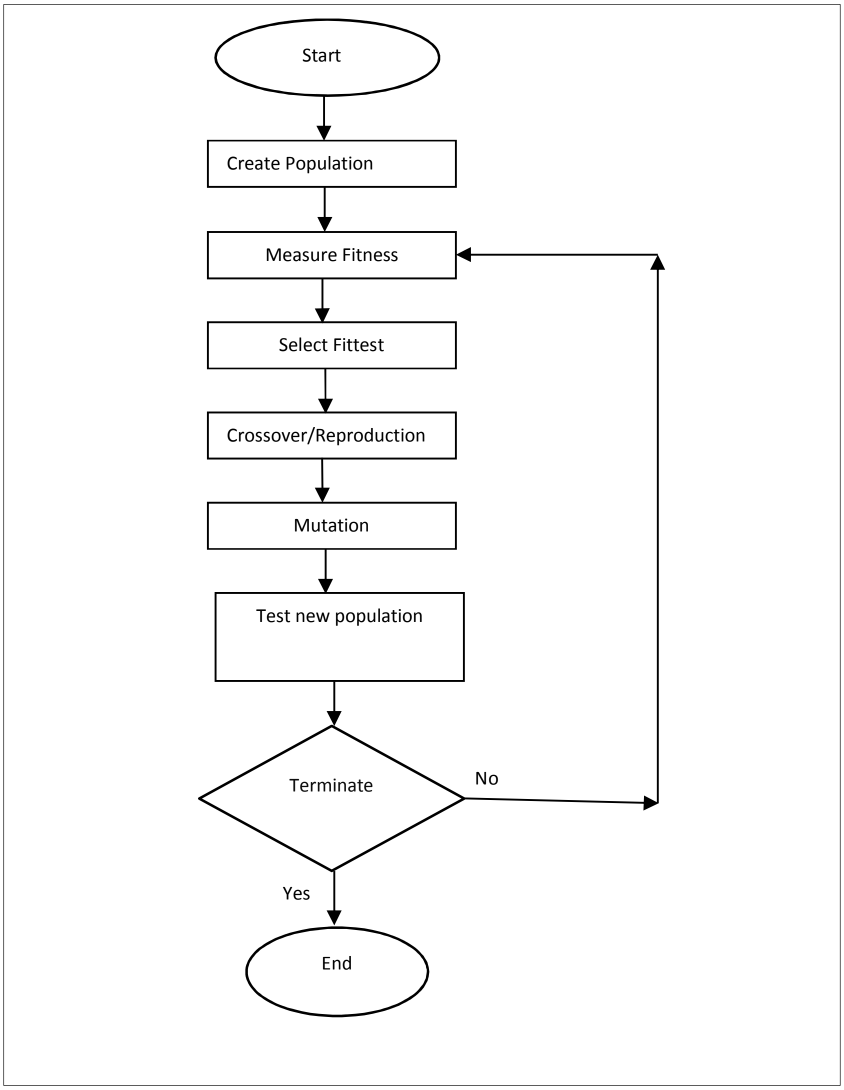


Figure 2.8: Genetic Algorithm process

2.2.2.5 Particle Swarm Optimisation

Particle Swarm Optimisation (PSO) is an evolutionary algorithm developed by Eberhart and Kennedy [40]. It is a population based method which utilises swarm intelligence generated by cooperation and competition between particles in a swarm [11]. These particles change their position with time in a defined search space according to its own experience as well as the experience of other neighbouring particles. A PSO thus combines both local and global search methods [12].

The PSO process involves the updating of a particles velocity and position with time until the best solution is obtained. These two variables are adjusted using the following equations:

$$V_i = \omega * V_{i-1} + c_1 * rand(0,1) * (p_{best} - p_{location}) + c_2 * rand(0,1) * (g_{best} - p_{location}) \quad (2.2)$$

$$p_{location} = prv_{location} + V_i \Delta t \quad (2.3)$$

where:

V_i is the current velocity of particle i

Δt represents the time interval

ω represents the inertia weight

V_{i-1} is the previous velocity

Rand (0,1) are random values between 0 and 1

p_{best} is the particles best personal position

$p_{location}$ is the particles present location

$prv_{location}$ is the previous location of the particle

g_{best} is the best value obtained by any particle in a swarm with reference to its neighbours

$c_1$and c_2 are the learning and acceleration coefficients. The typical values for these coefficients are 2 for both variables.

Quaiyumn *et al.* [12] and Shayeghi *et al*. [11] both proposed the use of PSO with an Artificial Neural Network. PSO in both cases was used to evolve the neural network weights in order to obtain the best set of weights that result in a low forecasting error. Shayeghi *et al.* proposed the clustering of data into special days and normal working days based on correlation studies conducted between load and various load affecting factors such as holidays and weather variations. The authors indicate that there is an improvement in speed of convergence as well as accuracy of load forecasts when PSO is used in conjunction with ANNs.

A Alshareef [14] presents a hybrid load forecasting model for the Western area of Saudi Arabia consisting of PSO and ANN. The author first conducted forecasts with an ANN and thereafter used PSO to further optimise the results. It was found that the forecasting errors obtained for the ANN were high as compared to the hybrid method. The results of the hybrid model range between 0.6% and 4.80%.

The pseudocode for PSO can be written as follows [41]:

```
For each particle
        Initialize particle
end

Do
For each particle
        Calculate fitness value
        If the fitness value is better than the best fitness value(pBest) in history
                Set current value as the new pBest
        end
        Choose the particle with the best fitness value of all the particles as the gBest
end
For each particle
        Calculate particle velocity according to equation (2.2)
        Update particle position according to equation (2.3)
end

While maximum iterations or minimum error criteria is not attained
```

Figure 2.9: Pseudocode for PSO

From the literature that was surveyed, it can be deduced that hybrid methods produce better results as compared to single methods. This principle will be investigated in this research by using PSO, with ANN. It was found that the factors affecting the load such as humidity, wind speed, rainfall, etc., need to be incorporated into the STLF model to improve accuracies whenever possible. It is pertinent to conduct a correlation study between the electrical load and the weather elements to determine the variables with the most significant impact. In this way, one can avoid utilising variables that have little to no impact on the load profile. This research uses the correlation method in order to determine the load affecting variables. Alshareef in [14] utilises average temperature with historic load data and also groups the data into seasonal clusters in order to perform his forecast. In this research, an added variable (humidity) is used as well as the actual minimum and maximum temperatures for the forecast and previous days.

Forecasting for day types is an avenue that most literature explores. Days are split into categories such as weekends and weekdays or each day is a category on its own however this is dependent on the load shapes. Forecasting for holidays and special events however has not been widely researched; very few models have been developed for these types of days. This is an area that requires more attention as there is a need to accurately forecast for these days in order to avoid over or under-committing on generation.

This research undertakes to include the load affecting variables as input to the load forecast. One of the gaps that this research will investigate is the determination of the most important factors that affect load forecasting. The back propagation training that is generally used for training ANNs is also found to be an area that can be improved due to the slow convergence problem as well as the issue of settling into local minima. An investigation into the use of PSO, to mitigate against these issues, is conducted.

Chapter 3 : Short Term Load Forecasting using Artificial Neural Networks

This section discusses the use of ANN networks for the problem of short term load forecasting. The following methods are investigated: Feed-forward ANN and Elmans Recurrent NN (ERNN). These methods are then utilised to create models which are then tested using Matlab neural network toolboxes and the results compared to determine the best performing ANN. A comparison with a traditional forecasting technique, namely, Multiple linear regression, is performed.

3.1 Forecasting using Feed-Forward ANN models

The connection types for feed-forward ANN is static since it merely accepts an input (x) and synthesises it with the weights (w) through an activation function (f) to produce an output (y). This can be seen in Figure 3. 1.

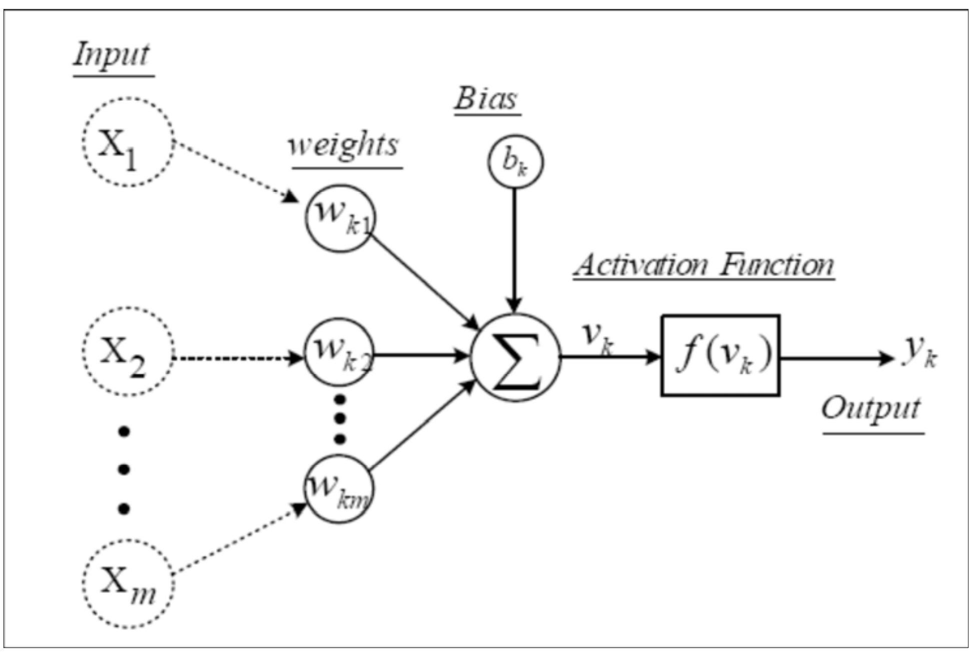


Figure 3.1: Structure of a neuron [42]

The topology for these networks is generally a single layer neural network or a multi-layer network made up of an interconnection of neurons such as those depicted in Figure 3.1. The difference between the two is that the single layer network has one layer of neurons with no activation function while a multi-layered network has more than one layer of neurons and uses activation functions. The single layer neuron is not adequate for load forecasting due to the complex nature of the forecasting problem. For an adequate prediction, a multi-layer neural network is more preferable as it can be able to easily extract the relationships between the input and output variables. An example of a multi-layer network is shown in Figure 3.2 below, this network has 2 layers, an input layer as well as a hidden layer and is this called a 2-layer feed-forward neural network.

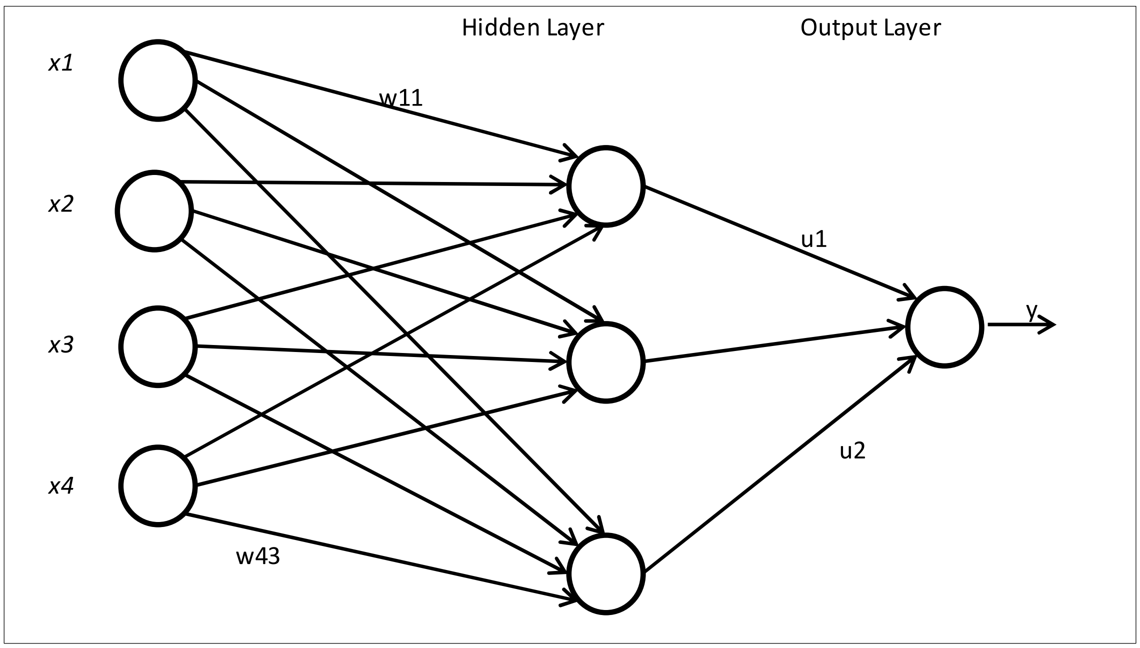


Figure 3.2: Multi-layer neural network [14]

An activation function performs a mathematical operation of the output of a signal as shown in Figure 3.1. The activation or transfer functions commonly used with neural networks are shown in Figure 3.3 and are chosen based on the type of problem that needs to be solved [43]. Each of the transfer functions is characterised as follows [44, 45]:

1. **Threshold function (step function):** This function produces a positive output value only which is either 0 or 1 over the range of input values [-∞, ∞]. This is generally used for the input and output layer neurons.

2. **Signum function:** This function is the same as the threshold function with the difference being the output value lies between [-1, 1] over the input range [-∞, ∞]. For a positive input, the output of the function is a positive 1 while for a negative input; the output of the function is a negative 1.

3. **Piece-wise linear function (saturation):** This function can have a binary or bipolar range for the saturation limits. It is a mix of a linear function, where the output is equal to the input for the sloped section of the graph (see Figure 3.3), and a signum function.

4. **Logistic function (monopolar sigmoid):** This is a smoothed version of the signum or threshold function. It is of a differentiable form and is most commonly used for classification and load forecasting problems [44, 46]. It produces an output between [0,1] for inputs over the range [-∞, ∞].

5. **Hyperbolic tangent function (bipolar sigmoid):** This function is also one of the most commonly used activation functions [44]. It differs with the logistic function in that it produces an output over the range [-1, 1] for inputs between [-∞, ∞].

Once a network topology has been decided upon (i.e., the number of layers the network will have, the type of activation functions used for the various layers), this network will need to be trained so that it is able to detect a pattern given a set of inputs and is able to forecast as required. In this study, the commonly used hyperbolic tangent function was used. The next section will discuss the training algorithm that has been used in this study and provides detail on how it is used within the neural network.

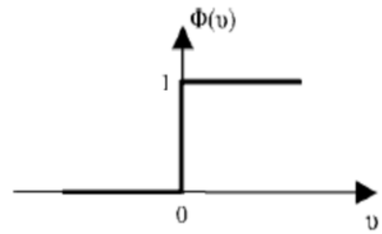
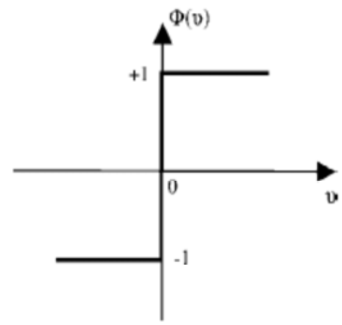
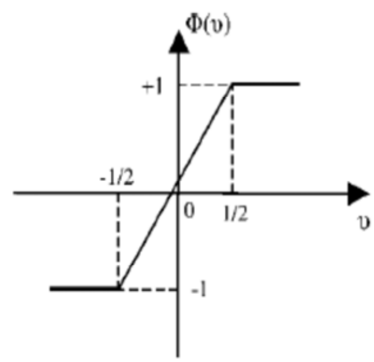
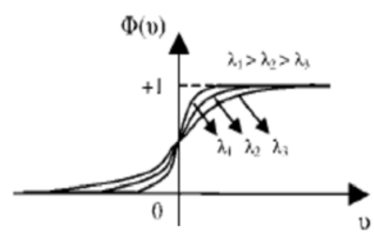
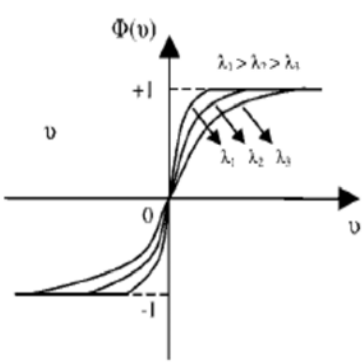
	Function's name	Function's definition	Pictorial representation
1	Threshold function (step function)	$\Phi(\upsilon)=\begin{cases}1, & \upsilon \geqslant 0\\ 0, & \upsilon < 0\end{cases}$	
2	Signum function	$\Phi(\upsilon)=\begin{cases}1, & \upsilon > 0\\ 0, & \upsilon = 0\\ -1, & \upsilon < 0\end{cases}$	
3	Piece-wise linear (saturation) function	$\Phi(\upsilon)=\begin{cases}-1, & \upsilon \leqslant \frac{1}{2}\\ \upsilon, & -\frac{1}{2} < \upsilon < \frac{1}{2}\\ 1, & \upsilon \geqslant \frac{1}{2}\end{cases}$	
4	Logistic function (monopolar sigmoid)	$\Phi(\upsilon)=\frac{1}{1+\exp(-\lambda\upsilon)}$	
5	Hyperbolic tangent function (bipolar sigmoid)	$\Phi(\upsilon) = \tanh\left(\frac{\lambda\upsilon}{2}\right) = \frac{1-\exp(-\lambda\upsilon)}{1+\exp(-\lambda\upsilon)}$	

Figure 3.3: Types of activation functions [47]

3.1.1. Back Propagation Method

Training is the process by which an ANN determines the different network parameters such as weights and biases which produce an optimal result. Training of neural networks can be classified into two categories, i.e., supervised and unsupervised learning. The difference between the two is that during supervised learning there is a target output with which the network can compare its output against and can thereby adjust weights until convergence is reached, it is a static method whereas the unsupervised learning method does not have a target output. In unsupervised learning, the network is presented with a set of inputs which it will use to develop a pattern. This is generally known as self-organizing map.

A training data set needs to be provided to the neural network in order for it to begin determining the optimal network parameters. This data set needs to cover a wide range of input patterns which should be sufficient enough for the network to recognize and predict the relationship between input variables and target output [29]. One of the supervised learning algorithms used for ANN training is the back propagation. A multi-layered neural network trained by the back propagation method (BP) is the most common architecture and has been applied to a wide variety of ANN problems of which STLF is included. The generalized delta rule is applied to adjust the weights of the feed- forward networks thus minimizing a predetermined cost error function. In order to illustrate the BP method, consider a multi-layer feed- forward network shown in Figure 3.4. It consists of two inputs X1 & X2 and one output O.

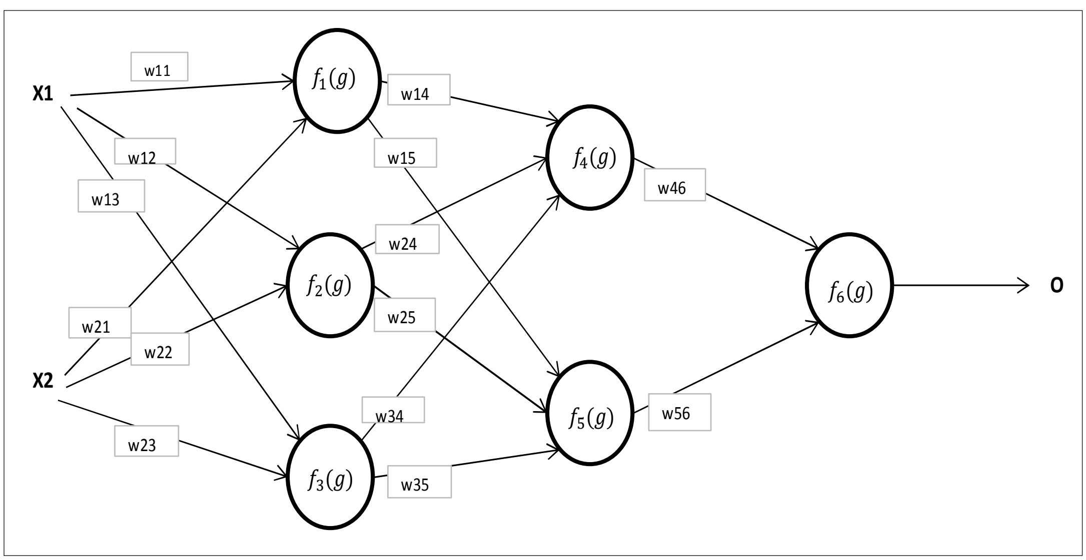


Figure 3.4: Multi-layer feed-forward network

The output of each neuron is given by the following equation:

$$y_1 = f_1(w_{11}x_1 + w_{21}x_2) \tag{3.1}$$

y_1 represents the output of the first neuron in Figure 3.4 and f_1 represents the transfer function of the first neuron which takes in the summated values of the product of the weights (w) with the inputs (x). This equation is used for all the outputs of the neurons of the first layer i.e. y_2 and y_3.

The outputs of the neurons in the second layer are given as follows:

$$y_4 = f_4(w_{14}y_1 + w_{24}y_2 + w_{34}y_3) \tag{3.2}$$

The final value at the output layer would then be given by

$$o = f_6(w_{46}y_4 + w_{56}y_5) \tag{3.3}$$

Once the outputs have been computed, this value is then compared to the target value t and the difference between the two is called an error signal and it is given by *d* as illustrated in Figure 3.5.

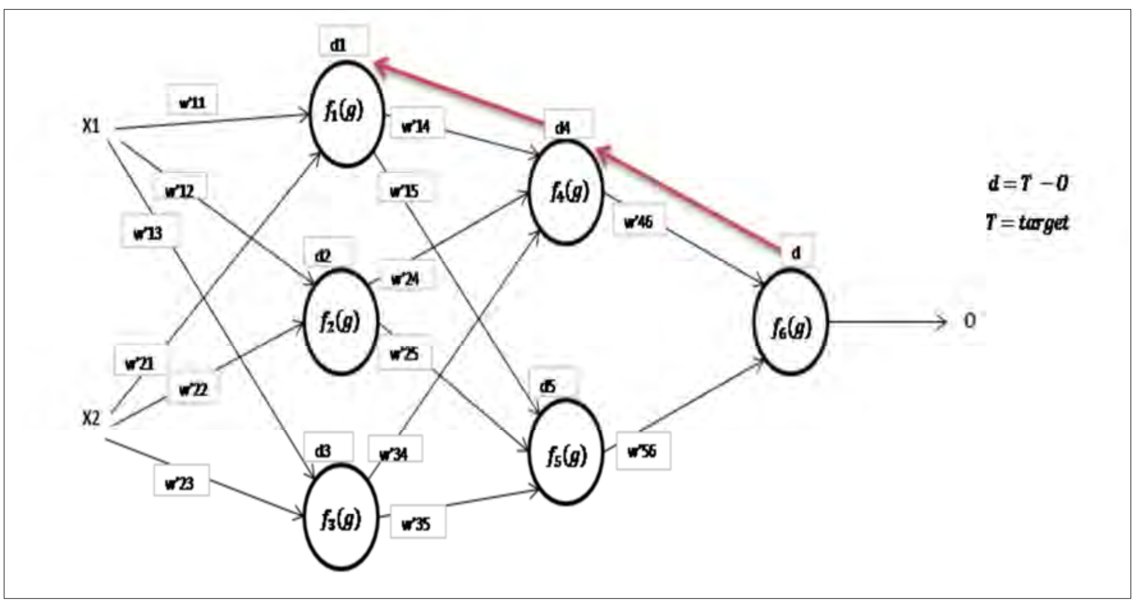


Figure 3.5: back propagation of error signal

This error signal has to be propagated back through the network from the output layer to the input layer in order to modify the weights in an iterative manner until the desired error goal is reached.

The final error signal is as follows

$$d = T - O \tag{3.4}$$

where T is the target and O is the predicted output.

The error signal needs to be computed for all neurons in the network for example the error signal d_4 is a product of the weight connected to the output neuron with the final error signal d (see Fig. 3.5).

$$d_4 = w_{46}d \tag{3.5}$$

$$d_1 = w_{14}d_4 + w_{15}d_5 \tag{3.6}$$

The weights of all the layers would then be modified as follows once the error signals for all neurons have been calculated.

$$w'_{11} = w_{11} + \alpha d_1 \frac{df_1(g)}{dg} X_1 \tag{3.7}$$

$$w'_{21} = w_{21} + \alpha d_1 \frac{df_1(g)}{dg} X_2 \tag{3.8}$$

In equations (3.7) & (3.8), α is the learning rate and w'11 and w'21 are the weights connecting the inputs to the first layer of neurons.

The data provided to neural networks for training need to be carefully chosen and pre-processed so as to make it easy for the neural networks to make the correct associations. The next section discusses the steps that can be taken to ensure that data provided to the neural networks are sufficient.

3.1.2 Input vectors

Selection of the kind and number of input variables is an important issue encountered in the design of the ANN. In case of STLF, the performance of the forecast depends largely on the proper selection of the load affecting variables. In STLF, the key variables are time, forecasted weather variables, and historical load. Therefore, it is vital to identify the input variables that have significant impact on the system load.

This is particularly important since inclusion of irrelevant inputs or inputs with no significant impact on the target outputs can distort the forecast performance, increase the training time, increase network complexity and reduce the network execution time. One approach to identify the most affecting input variables is by evaluating the statistical correlation between such input variables and the target output [29]. Once the selection has been conducted, data will need to be normalised. The following method is generally used to normalise the data:

$$l_s = \frac{l - l_{min}}{l_{max} - l_{min}} \tag{3.8}$$

where l_s is the scaled value and l_{min} and l_{max} being the minimum and maximum values of the variables. The scaled values of the selected variables (inputs and outputs) can then be used to conduct a load forecast.

3.2 Proposed Feed-Forward model

This section discusses the model that was developed and tested using Matlab version 7.0.

3.2.1 Data selection

The data set used for this analysis was obtained from Eskom Distribution. The area of study is a mainly residential area in the province of Kwa-Zulu Natal (KZN) for a 132/11 kV substation called Abattoir. Weather data for the area was obtained from the South African Weather Services (SAWBS). The data sets for both weather and historical load were obtained for the period from 2010 – 2011. Data from the year 2010 was used to train and validate the neural network. Select days in the year 2011 were then used to test the performance. Holidays were removed from the data set as holiday forecasting was not within the scope of this study. The data was split into weekday and weekend loads as these loads did not have similar load profiles. Two forecasters were thus used for simulation purposes.

A correlation study to determine the weather variables that play a significant role in influencing the loading of the substation was conducted. The following variables were obtained from SAWBS: hourly humidity and temperature values as well as monthly rainfall measurements. Rainfall was removed from the correlation study and forecasting, as some of the data for most months/days was incomplete or missing. Rainfall as a weather component plays a role in determining the load profile however the degree of influence would have to be determined by using a correlation study. The lack of complete data was a limitation for this analysis. The results of the correlation analysis contained in Table 1 below shows that there is a strong linear relationship between load and temperature as well as between load and humidity. These parameters were included as part of the input vector to the neural networks.

Correlation coefficients (r) range from -1 to 1 [48]:

- $r = +1$ represents a perfect linear correlation with the variables
- $0<r<0.09$ represents no correlation
- $0.1 <r<0.25$ represents a small linear correlation
- $0.26 <r<0.40$ represents a medium linear correlation
- $0.4<r<1.0$ represents a strong linear correlation
- $r = 0$ means that the variables are not related
- $-1<r<0$ represents a negative linear correlation (a negative correlation is the relationship between variables such that as one value increases, the other decreases)
- $r = -1$ represents a perfect negative linear correlation

Table 3.1: Correlation Analysis

Variable	Load
Temperature	**0.667544039**
Humidity	**0.563689217**

Table 3.1 above shows that there is a strong linear relationship between load and temperature as well as between load and humidity. These weather variables were thus included as part of the input vector to the neural networks. The data were normalised using equation (3.8). The correlation study was based on weather variables only even though there are other factors that influence the load profile such as day type, day of the week. These variables were included as inputs without a correlation study based on the information obtained while conducting the literature review in chapter 2.

3.2.2 ANN Architecture

Table 3.2 describes the inputs and outputs for three ANN models (ANN, ANN-t and ANN-w) that were developed. ANN considers historical load only, ANN-t considers historical load and temperature and ANN-w considers historical load, temperature as well as humidity as inputs. This was done in order to ascertain the significance of the weather variables. Although a correlation study was conducted, the two ANN's were created in order to validate the hypothesis that the inclusion of weather variables has a positive effect on the accuracy of a forecast.

Table 3.2: ANN models used for forecasting

Models	Input	Description	Output	Description
ANN	1-48	Previous day half-hourly load data	1-48	Forecasted half hourly data
	49	Previous day type		
	50	Forecast day type		
ANN-t	1-48	Previous day half-hourly load data	1-48	Forecasted half hourly data
	49-50	Previous day min and max Temperature		
	51-52	Forecast day min and max Temperature		
	53	Previous day type		
	54	Forecast day type		
ANN-w	1-48	Previous day half-hourly load data	1-48	Forecasted half hourly data
	49-50	Previous day min and max Temperature		
	51-52	Forecast day min and max Temperature		
	53-54	Previous day min and max Humidity		
	55-56	Forecast day min and max Humidity		
	57	Previous day type		
	58	Forecast day type		

A Netlab toolbox, developed by Ian Nabney [49], was used to train and test the ANN models. The toolbox uses a training algorithm called the scaled conjugate gradient (SCG) back propagation method.

Traditional BP is a gradient descent local search procedure that measures the output error, calculates the gradient of the error by adjusting the weights in the descending gradient direction whereas SCG is a second order algorithm in the conjugate gradient methods [50]. In these methods, a search is performed along conjugate directions, which produces generally faster convergence than steepest descent directions. SCG does not require line search at each iteration step like other conjugate training functions. Step size scaling mechanism is used which avoids a time consuming line search per learning iteration. This mechanism makes the algorithm faster than any other second order algorithms [50]. SCG was thus chosen due to its ability to converge well for a network with a large number of weights.

3.2.3 Training of ANN Models

Two forecasters (Table 3.3 and 3.4) were tested i.e. weekend and weekday forecasters with the 3 model variation as discussed in section 3.2.2. The number of hidden layer neurons was determined by trial and error by looking at the topology that provides a minimum training error. From the literature survey that was conducted, it was found that one of the difficulties of a feed-forward ANN is that there are no mathematical rules available to calculate the number of hidden layers or number of hidden layer neurons required to provide an optimal network [51]. The neural network design was chosen by varying the number of hidden layer neurons until an optimal performance error was obtained. Table 3.3 shows the error obtained with the various hidden layer neurons. The training error was evaluated by looking at the lowest Mean Square Error (MSE) which is given by the following equation:

$$\text{MSE} = \frac{1}{N} * \sum_{i=1}^{N} (L_{ai} - L_{fi})^2 \tag{3.10}$$

Where N is the total number of half hourly load points in a day, L_{ai} is the actual load at point *i* and L_{fi} is the forecasted load at point *i* in a day.

The activation functions for the models were chosen as follows:

1. Input layer – linear
2. Hidden layer – tangent sigmoid
3. Output layer – linear

Table 3.3: Final ANN architectures for Weekday forecaster

Forecaster	Models	Inputs	Hidden Neurons	MSE	Models	Inputs	Hidden Neurons	MSE	Models	Inputs	Hidden Neurons	MSE
WEEKDAY	ANN	50	5	11.04	ANN-t	54	5	7.81	ANN-w	58	5	8.52
			10	9.6			10	6.83			10	6.51
			15	7.99			15	5.58			15	6.02
			20	7.93			20	5.38			20	4.61
			25	7.47			25	5.03			25	4.72
			30	7.53			30	4.5			30	4.33
			35	6.88			35	4.28			35	4.56
			40	6.81			40	4.33			40	4.2
			45	6.8			45	4.2			45	3.97
			50	6.29			50	4.21			50	3.77
			55	6.62			55	4.17			55	3.81
			60	6.61			60	3.96			60	3.46
			65	6.69			65	3.99			65	3.5
			70	6.71			70	4.05			70	3.65
			75	6.25			75	3.83			75	3.5
			80	6.41			80	3.85			80	3.42
			85	6.86			85	3.71			85	3.24
			90	6.15			90	3.8			90	3.36
			95	6.45			95	3.9			95	3.7
			100	6.34			100	3.88			100	3.33

Table 3.4: Final ANN architectures for Weekend forecaster

Forecaster	Models	Inputs	Hidden Neurons	MSE	Models	Inputs	Hidden Neurons	MSE	Models	Inputs	Hidden Neurons	MSE
WEEKEND	ANN	50	5	1.62	ANN-t	54	5	1.46	ANN-w	58	5	1.36
			10	1.63			10	1.01			10	0.79
			15	0.99			15	0.66			15	0.89
			20	0.82			20	0.6			20	0.59
			25	0.79			25	0.58			25	0.57
			30	0.73			30	0.56			30	0.5
			35	0.79			35	0.48			35	0.38
			40	0.76			40	0.45			40	0.35
			45	0.69			45	0.4			45	0.39
			50	0.72			50	0.39			50	0.31
			55	0.67			55	0.42			55	0.36
			60	0.58			60	0.41			60	0.29
			65	0.68			65	0.47			65	0.34
			70	0.61			70	0.35			70	0.23
			75	0.58			75	0.38			75	0.25
			80	0.56			80	0.36			80	0.24
			85	0.54			85	0.36			85	0.2
			90	0.65			90	0.43			90	0.27
			95	0.65			95	0.42			95	0.25
			100	0.57			100	0.4			100	0.26

From Tables 3.3 and 3.4, it can be seen that the number of hidden layer neurons ranged between 70 and 90. The number of hidden layer neurons highlighted in the tables produced the lowest error during training. These neurons were included in the ANN networks and were used for the testing phase of the study which is discussed in the next section.

3.2.4 Simulation Results

The following Figures 3.6 to 3.18 illustrate the results obtained from the simulation of weekdays and weekends in the year 2011. The ANN networks were tested using the same test data. To quantify the performance of each ANN model, a mathematical approach known as Mean Absolute Performance Error (MAPE) is used.

MAPE is given by the following equation:

$$\text{MAPE} = \frac{1}{N} * \sum_{i=1}^{N} \left| \frac{L_{ai} - L_{fi}}{L_{ai}} \right| * 100 \tag{3.11}$$

Where L_{ai} and L_{fi} are the actual and forecasted load, respectively. N is the number of data points which in this case is 48 since the forecast is for every half hour.

MAPE is regarded as the best performance calculator as compared to MSE in that it does not have a square term in its equation which accentuates the large errors. A small MSE value means that the model is stable however for large error terms in the data, the MSE results become misleading [52].

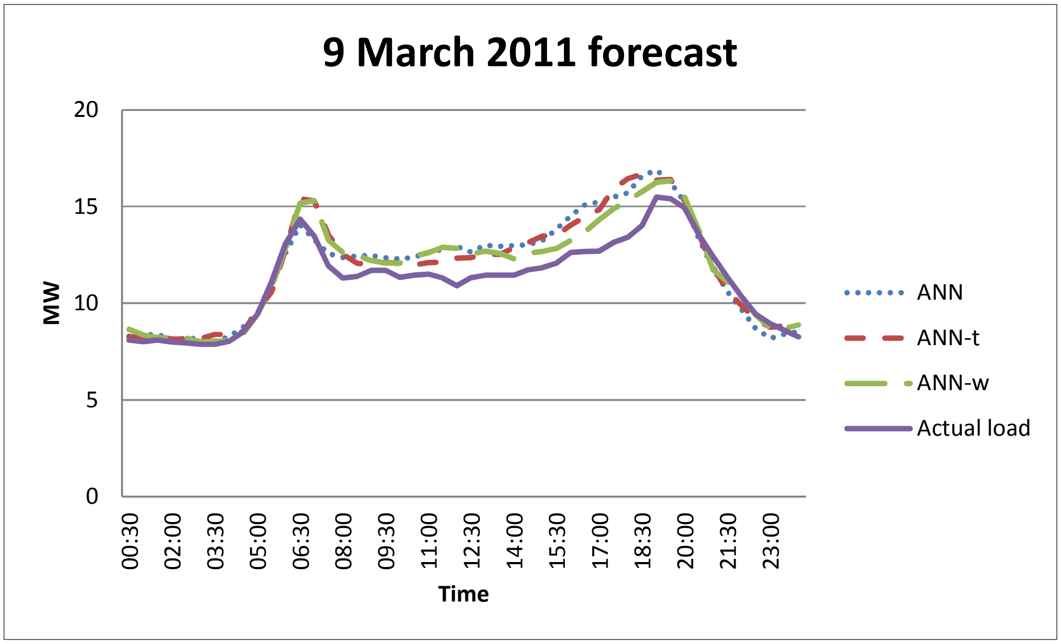


Figure 3.6: Wednesday March 2011 ANN load forecast

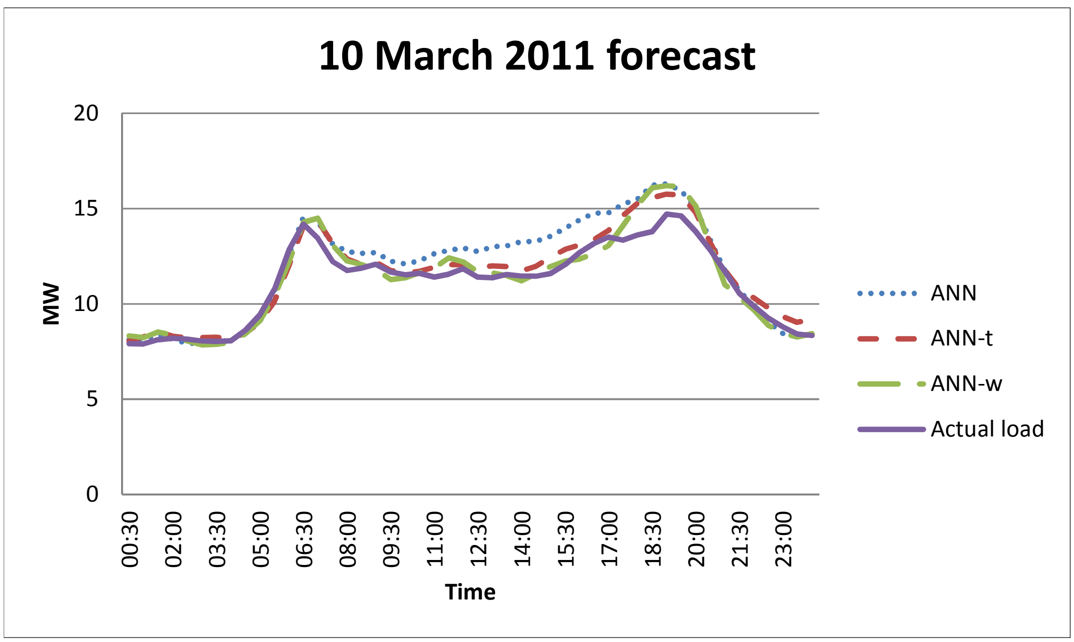


Figure 3.7: Thursday March 2011 ANN load forecast

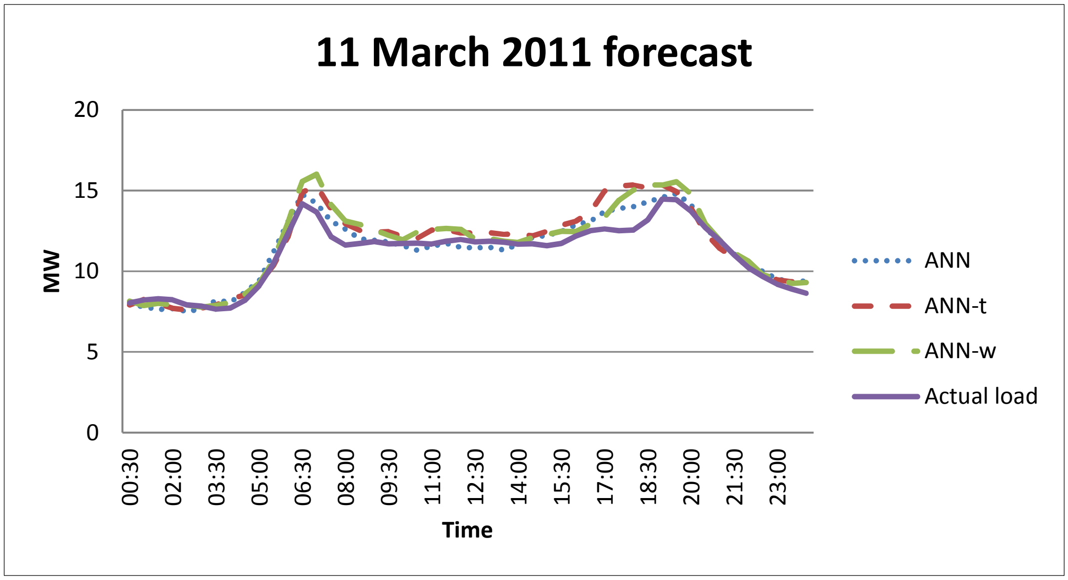


Figure 3.8: Friday March 2011 ANN load forecast

Table 3.5: ANN MAPE for weekdays in March 2011

Day	ANN	ANN-t	ANN-w
Wednesday 9 March 2011	8.05	7.46	6.35
Thursday 10 March 2011	6.91	4.40	3.93
Friday 11 March 2011	4.32	6.13	5.54

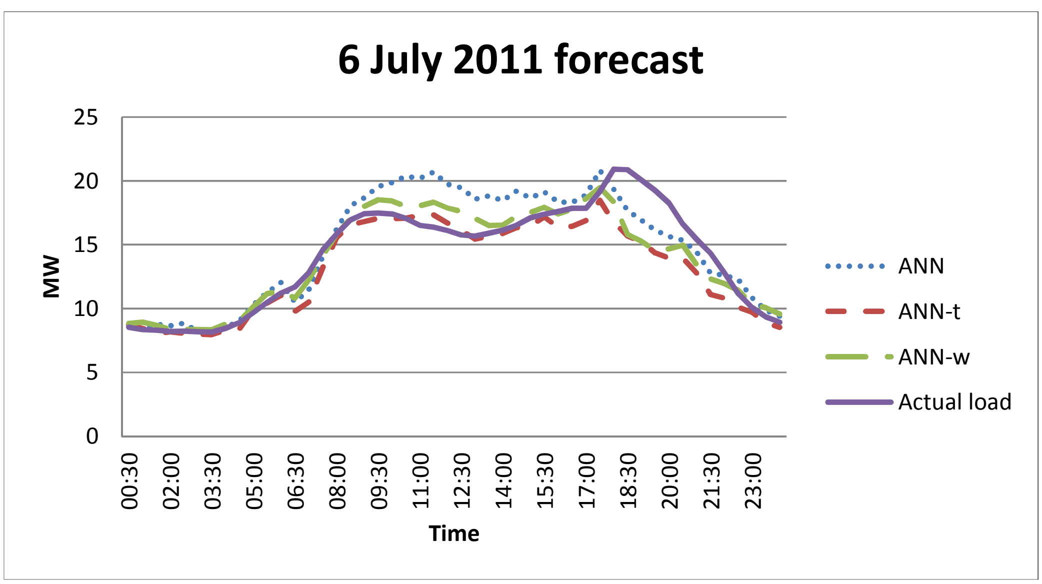


Figure 3.9: Wednesday July 2011 ANN load forecast

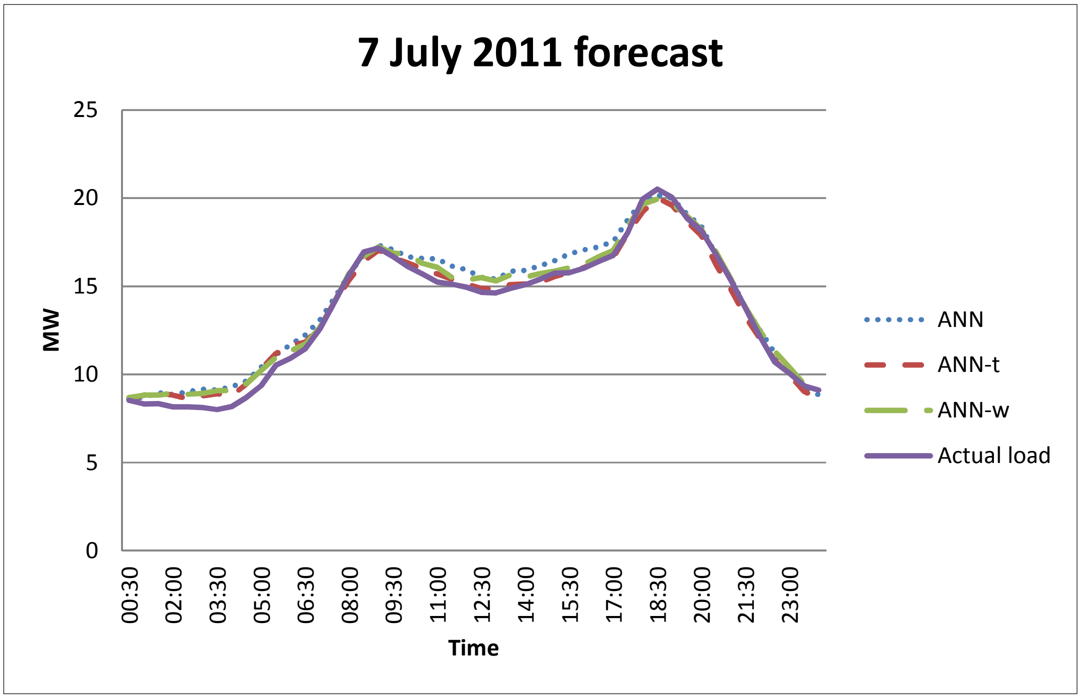


Figure 3.10: Thursday July 2011 ANN load forecast

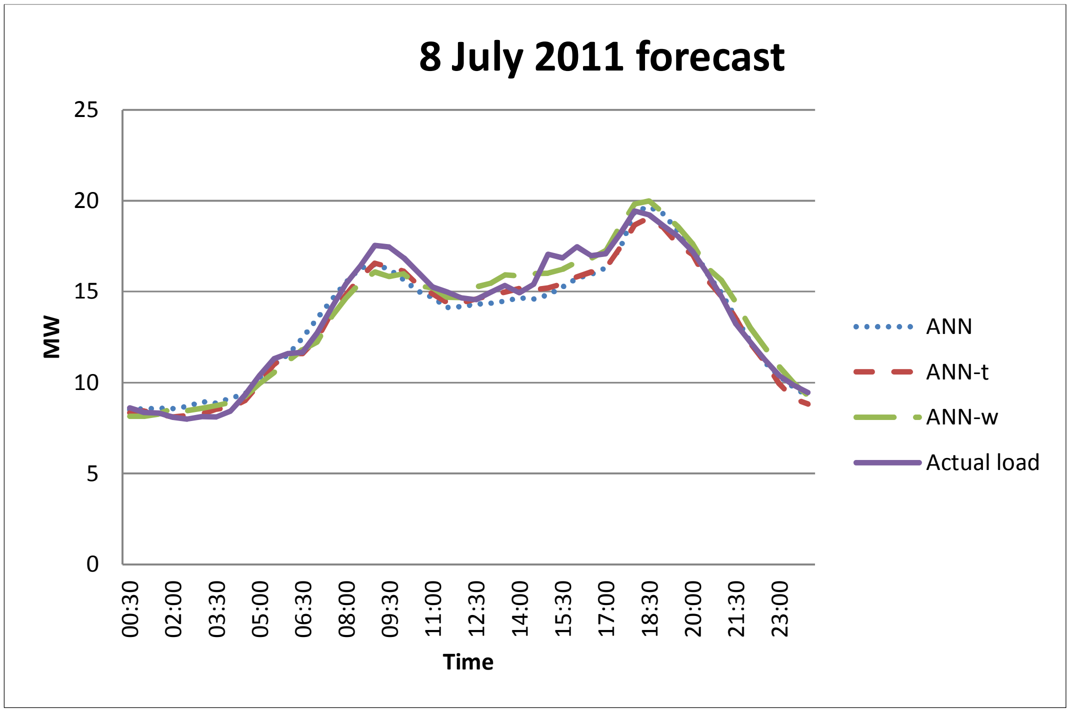


Figure 3.11: Friday July 2011 ANN load forecast

Table 3.6: ANN MAPE for weekdays in July 2011

Day	ANN	ANN-t	ANN-w
Wednesday 6 July 2011	9.52	7.18	6.69
Thursday 7 July 2011	4.88	3.05	3.55
Friday 8 July 2011	4.07	3.18	4.13

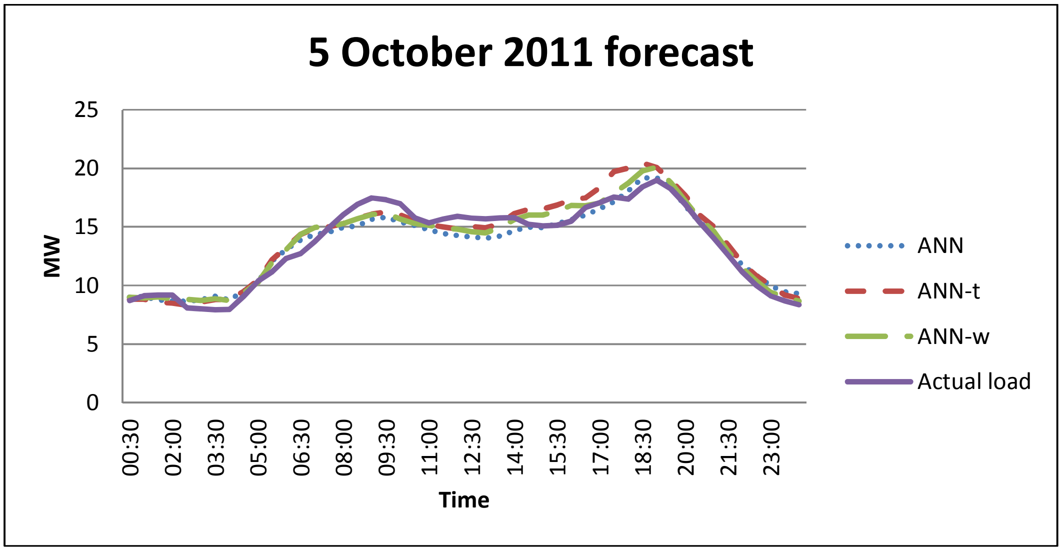


Figure 3.12: Wednesday October 2011 ANN load forecast

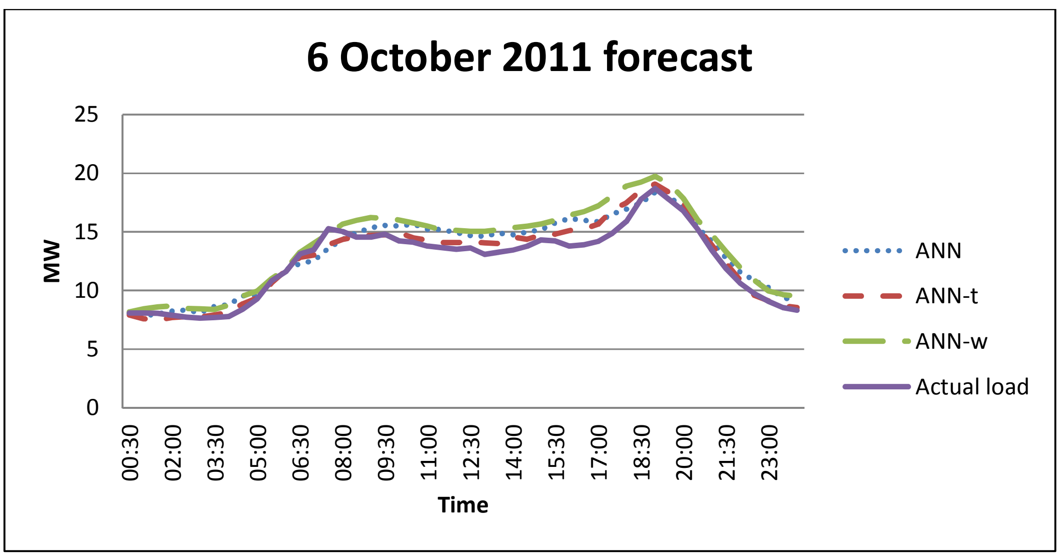


Figure 3.13: Thursday October 2011 ANN load forecast

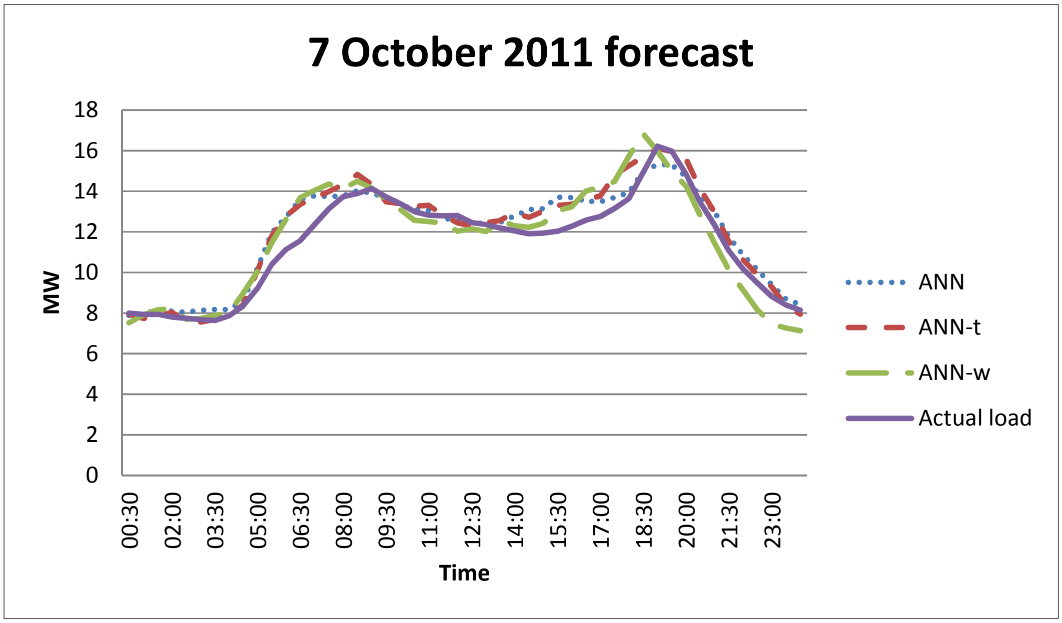


Figure 3.14: Friday October 2011 ANN load forecast

Table 3.7: ANN MAPE for weekdays in October 2011

Day	ANN	ANN-t	ANN-w
Wednesday 5 October 2011	5.77	6.59	5.08
Thursday 6 October 2011	7.55	4.09	10.33
Friday 7 October 2011	5.01	4.95	6.65

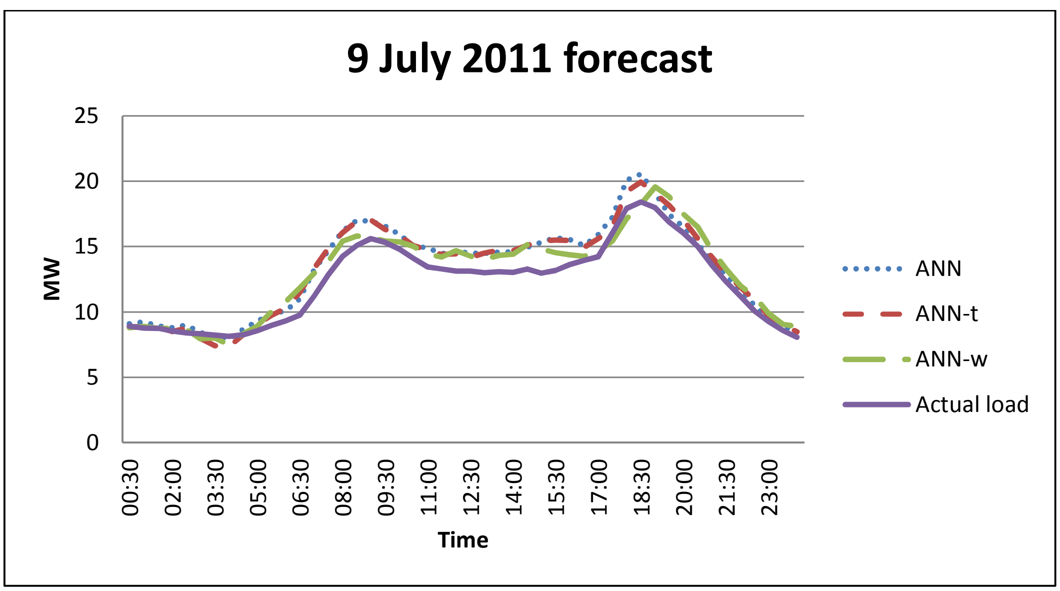


Figure 3.15: Saturday July 2011 ANN load forecast

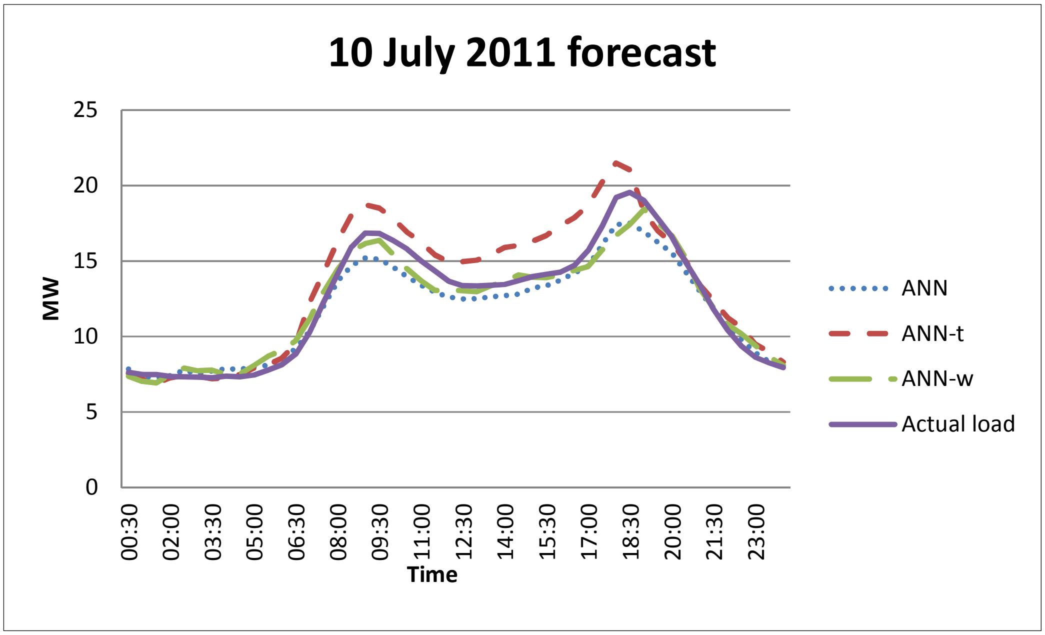


Figure 3.16: Sunday July 2011 ANN load forecast

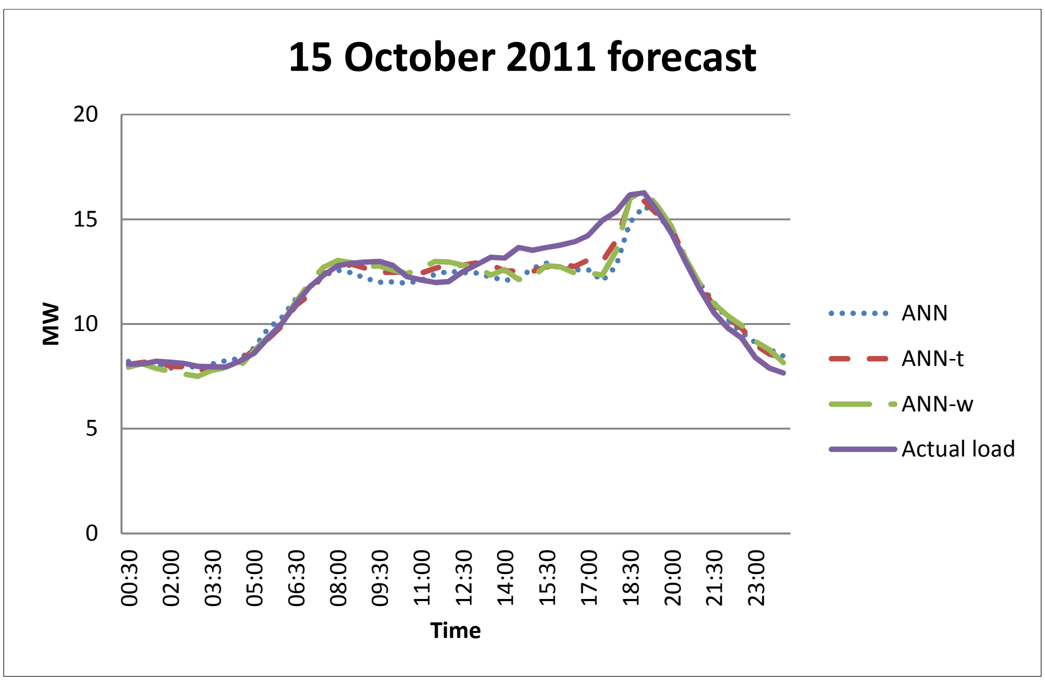


Figure 3.17: Saturday October 2011 ANN load forecast

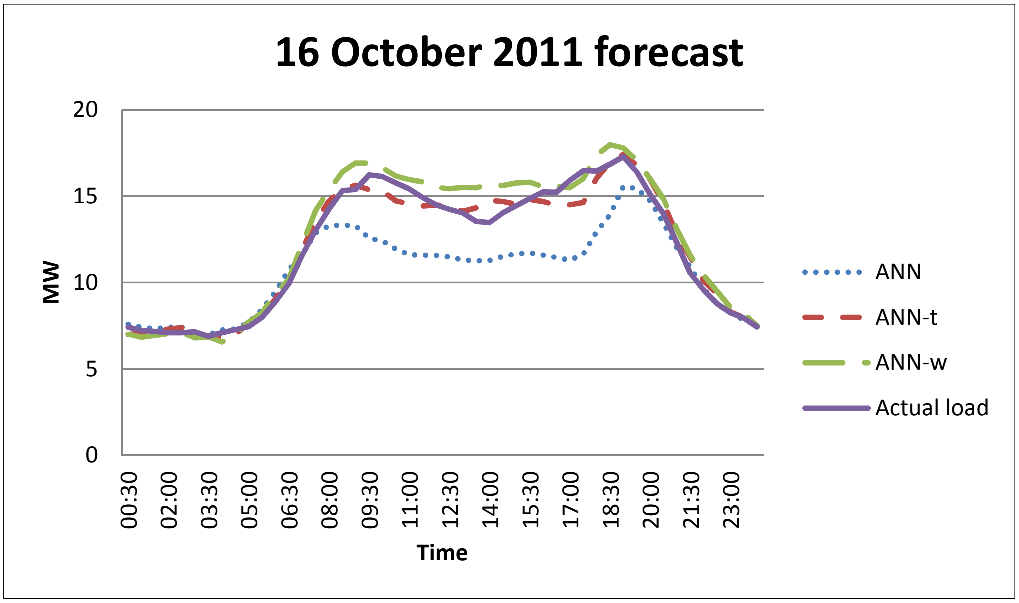


Figure 3.18: Sunday October 2011 ANN load forecast

Table 3.8: ANN MAPE for weekends in 2011

Day	ANN	ANN-t	ANN-w
Saturday 9 July 2011	8.08	8.27	7.12
Sunday 10 July	5.63	9.16	4.98
Saturday 15 October 2011	4.75	3.58	4.59
Sunday 16 October 2011	10.96	3.47	5.64

An analysis of the Figures 3.6 to 3.18 shows that each of the neural networks does very well in tracking the actual load profiles. The most important parts of the load profiles are generally the peak hours (morning and evening) which enable the generation dispatchers to dispatch power accordingly by ensuring that the scheduled generation will be able to supply the demand during peak hours. The best performing neural network was selected based on its performance during these hours as well as its MAPE. ANN-t tends to track the actual load profile much better than ANN-w and it is able to forecast the peak hour load quite well. ANN-w tends to overshoot or miss the peak hour load, particularly the evening peak. Table 3.9 shows the performance of all three networks.

Table 3.9: Performance evaluation of ANN networks using MAPE

Day	ANN	ANN-t	ANN-w
Wednesday 9 March 2011	8.05	7.46	6.35
Thursday 10 March 2011	6.91	4.40	3.93
Friday 11 March 2011	4.32	6.13	5.54
Wednesday 6 July 2011	9.52	7.18	6.69
Thursday 7 July 2011	4.88	3.05	3.55
Friday 8 July 2011	4.07	3.18	4.13
Saturday 9 July 2011	8.08	8.27	7.12
Sunday 10 July	5.63	9.16	4.98
Wednesday 5 October 2011	5.77	6.59	5.08
Thursday 6 October 2011	7.55	4.09	10.33
Friday 7 October 2011	5.01	4.95	6.65
Saturday 15 October 2011	4.75	3.58	4.59
Sunday 16 October 2011	10.96	3.47	5.64
Average MAPE	6.58	5.5	5.74

In Table 3.9, it shows that the network with the lowest performance error is ANN-t with a maximum of 9.16% and minimum of 3.05%. ANN-w also performed well with a maximum error of 10.33% and a minimum error of 3.93%. ANN performs consistently much poorer than ANN-t and ANN-w. The best performing network was thus selected as the ANN-t network which consists of temperature values as it had an average error of 5.5% while ANN and ANN-w had average errors of 6.58% and 5.74% respectively. This justifies the correlation results listed in Table 3.2 which shows that temperature has a stronger relationship with load as compared to humidity with load data. It can be deduced that humidity does have an influence on the load forecast for this geographical area of study however it is far outweighed by temperature, therefore for this research it can be removed as a load affecting variable. It is possible that for other geographical areas of study, humidity may play a vital role, therefore as discussed previously in this chapter; a correlation study is required so as to determine the most influential variables.

The next section discusses the use of a recurrent neural network for load forecasting instead of a feed-forward network. The reason the recurrent network was chosen was because of its speed of convergence. The ANN network proved to be very slow in converging, therefore the recurrent network was studied to determine whether the accuracy of the forecasts produced by the ANN could be improved. A brief overview is provided and the model used to conduct the simulations is presented.

3.3 Elman Recurrent Neural Network

Elman recurrent neural networks are feedback networks that are functions of both the current inputs as well as the previous outputs [18]. Recurrent ANNs are capacitated to internally encode temporal contexts from their feed-back connections. They evolve as a sequential system and, consequently, can describe a dynamical system evolution in a more efficient way than the feed forward models [53]. Several authors have indicated that the ERNN is more accurate than the various artificial neural networks available such as the multiple layer perceptron, radial basis networks etc. [21].

An ERNN is a feed-forward network that has the outputs of the hidden layer connected back to the inputs and is trained using a dynamic back-propagation training algorithm [21, 22]. Recurrent ANNs are capacitated to internally encode temporal contexts from their feed-back connections. They evolve as a sequential system and, consequently, can describe a dynamical system evolution in a more efficient way than the feed-forward models [21, 26]. Figure 3.19 illustrates an Elman RNN which is also known as a simple recurrent network (SRN) [24].

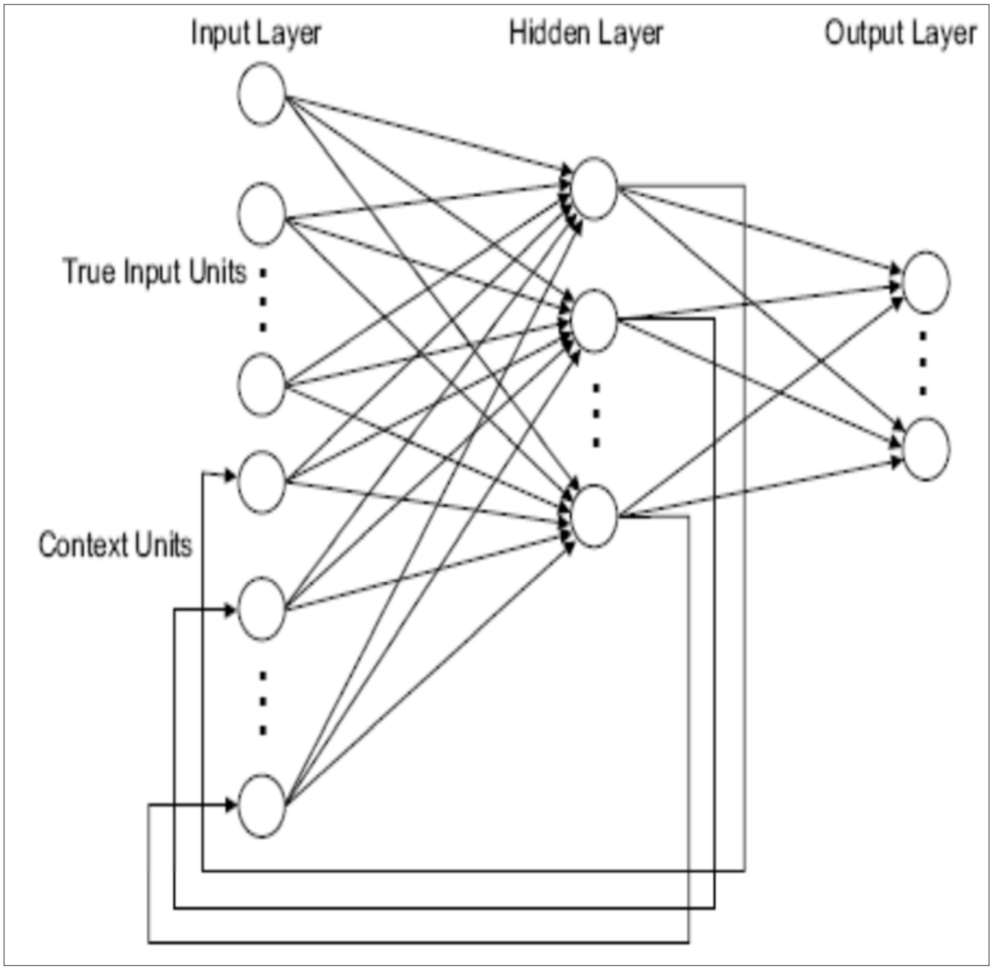


Figure 3.19: Elman recurrent neural network topology [24]

The following description of the way the ERNN operates is taken from [24]:

As depicted in Figure 3.19, the outputs of the hidden layer are fed back into the network through a context layer [27]. These are the only feedback connections in the network and the weights from the hidden layer to the context layer are constant values. All other connections are feed-forward with adjustable weights [20]. The number of context units is equal to the number of hidden layer neurons as can be seen in Figure 3.19. The Elman network has a large depth, low resolution memory, since the context units keep an exponentially decreasing trace of the past hidden neuron output values. In this network, signals are processed in two time steps. During the first step at time **t - 1**, signals from the input and context layers, which are fully connected to the hidden layer, are distributed to the hidden layer units. The pattern of activation outputs from the hidden layer are then computed and passed onto the output layer for processing at time **t**. At the same time, the hidden layer outputs are copied back onto a set of context units [24, 20].

Outputs from the context units then combine together with new input signals on the next cycle to feed the hidden units again at time **t+1**. Thus, the external inputs are being mixed with the previously computed inputs "in context" to give recurrent combinations of transformed inputs to the output layer. The weights on the feedback connections from the hidden to the context layer are fixed, typically as unit valued weights as they do not need to be trained, they merely refer the output of the hidden layer neurons to the context units in order to enhance the inputs. All other weights learn to encode sequences of input patterns during the training process.

The activation functions are non-linear differentiable functions (as discussed in section 3.1.1 under feed-forward neural networks) for the hidden layer neurons while the input neurons, context neurons and the output neurons have linear activation functions [24, 25, 20, 54]. As mentioned in previous sections, the training algorithm that is used for the ERNN is a back propagation algorithm that follows the method that was described in section 3.1.1. The only difference is that the context unit weights are not adjustable, they have a unit value weighting.

3.3.1 Proposed ERNN model

Table 3.10 depicts the two models designed for the purpose of this forecast study. ERNN is based on historical load and ERNN-w will account for the weather sensitive aspect of the data. The aim is to compare the performance of the networks when these variables are added.

Table 3.10: ERNN architecture

Models	Input	Description	Output	Description
ERNN	1-48	Previous day half-hourly load data	1-48	Forecasted half hourly data
	49	Previous day type		
	50	Forecast day type		
ERNN-t	1-48	Previous day half-hourly load data	1-48	Forecasted half hourly data
	49-50	Previous day min and max Temperature		
	51-52	Forecast day min and max Temperature		
	53	Previous day type		
	54	Forecast day type		
ERNN-w	1-48	Previous day half-hourly load data	1-48	Forecasted half hourly data
	49-50	Previous day min and max Temperature		
	51-52	Forecast day min and max Temperature		
	53-54	Previous day min and max Humidity		
	55-56	Forecast day min and max Humidity		
	57	Previous day type		
	58	Forecast day type		

3.3.2 Training of ERNN Models

The numbers of hidden neurons for both models were chosen based on trial and error method. The networks were trained by back-propagation algorithm. Table 3.11 and 3.12 lists the MSE for the various numbers of hidden neurons. The epoch was set to 10.

The activation functions for the models were chosen as follows:

1. Input layer – linear
2. Hidden layer – tangent sigmoid

3. Output layer – linear

It can be seen from Table 3.11 and 3.12 that all three ERNN models performed better with 20 hidden neurons. These neurons then made up the final topologies for the recurrent networks that were used for the simulation phases.

Table 3.11: Performance evaluation of ERNN models

Forecaster	Models	Inputs	Hidden Neurons	MSE	Models	Inputs	Hidden Neurons	MSE	Models	Inputs	Hidden Neurons	MSE
WEEKDAY	ERNN	50	5	0.0014	ERNN-t	54	5	0.0015	ERNN-w	58	5	0.0013
			10	0.0014			10	0.0014			10	0.0012
			15	0.0014			15	0.0013			15	0.0012
			20	0.0013			20	0.0011			20	0.001
			25	0.0013			25	0.0011			25	0.001
			30	0.0013			30	0.0012			30	0.001
			35	0.0013			35	0.0011			35	0.0011
			40	0.0013			40	0.0012			40	0.011
			45	0.0013			45	0.0012			45	0.001
			50	0.0013			50	0.0011			50	0.001

Table 3.12: Weekend ERNN network architecture

Forecaster	Models	Inputs	Hidden Neurons	MSE	Models	Inputs	Hidden Neurons	MSE	Models	Inputs	Hidden Neurons	MSE
WEEKEND	ERNN	50	5	0.002	ERNN-t	54	5	0.002	ERNN-w	58	5	0.002
			10	0.002			10	0.0019			10	0.0019
			15	0.0019			15	0.0019			15	0.0019
			20	0.0019			20	0.0019			20	0.0019
			25	0.0019			25	0.0019			25	0.0019
			30	0.0019			30	0.0019			30	0.0019
			35	0.0019			35	0.0019			35	0.0019
			40	0.0019			40	0.0019			40	0.0019
			45	0.0019			45	0.0019			45	0.0019
			50	0.0019			50	0.0019			50	0.0019

3.3.3 Simulation Results

The following Figures 3.20 to 3.32 illustrate the results obtained from the simulation of weekdays and weekends in the year 2011. ERNN networks were tested using the same test data.

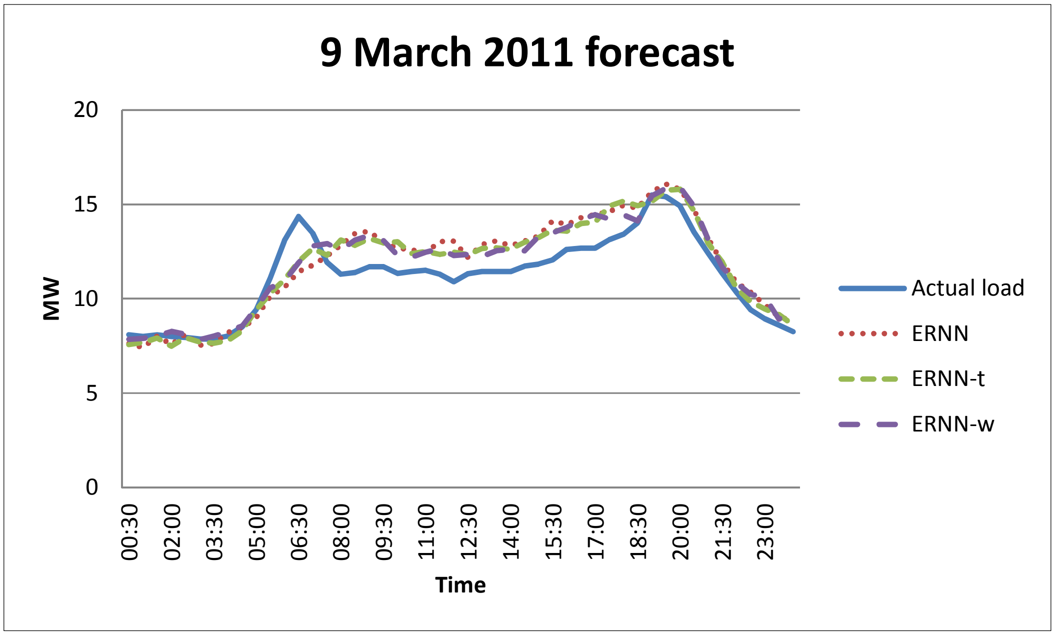


Figure 3.20: Wednesday March 2011 ERNN load forecast

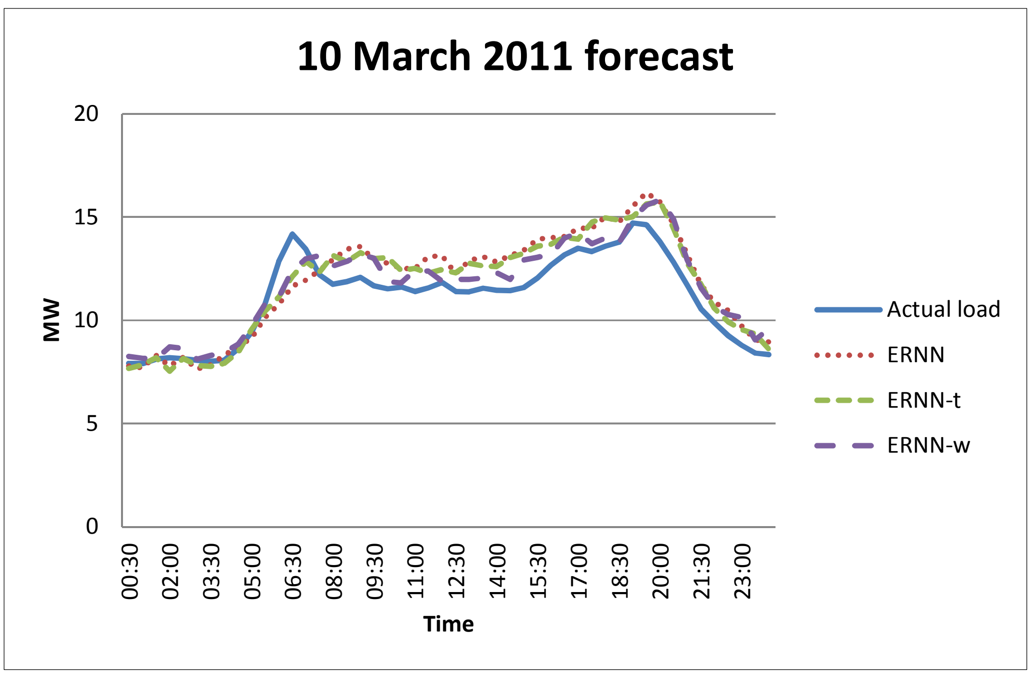


Figure 3.21: Thursday March 2011 ERNN load forecast

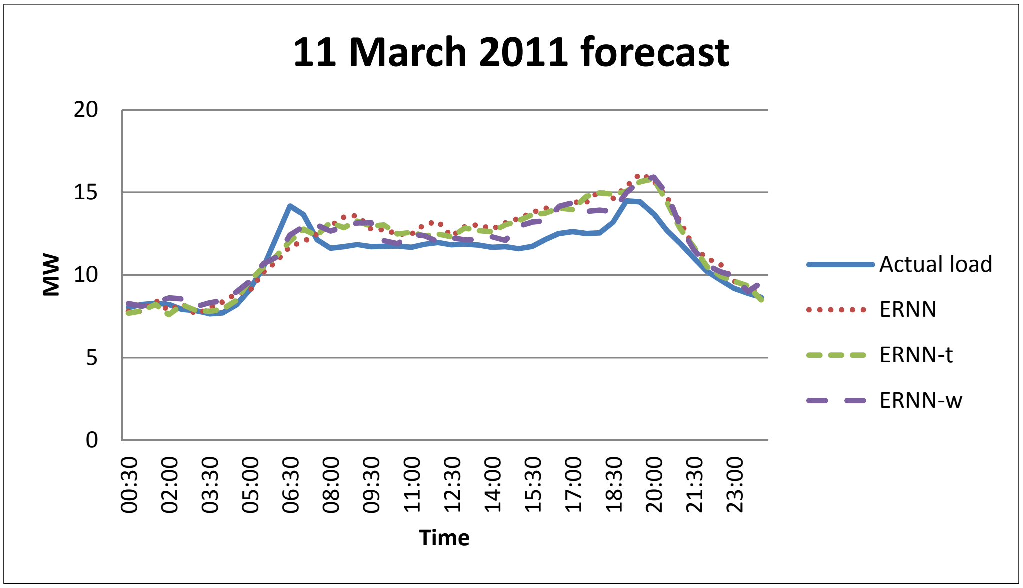


Figure 3.22: Friday March 2011 ERNN load forecast

Table 3.13: ERNN MAPE for weekdays in March 2011

Day	ERNN	ERNN-t	ERNN-w
Wednesday 9 March 2011	9.22	7.82	7.48
Thursday 10 March 2011	9.12	7.56	6.54
Friday 11 March 2011	9.25	7.83	7.10

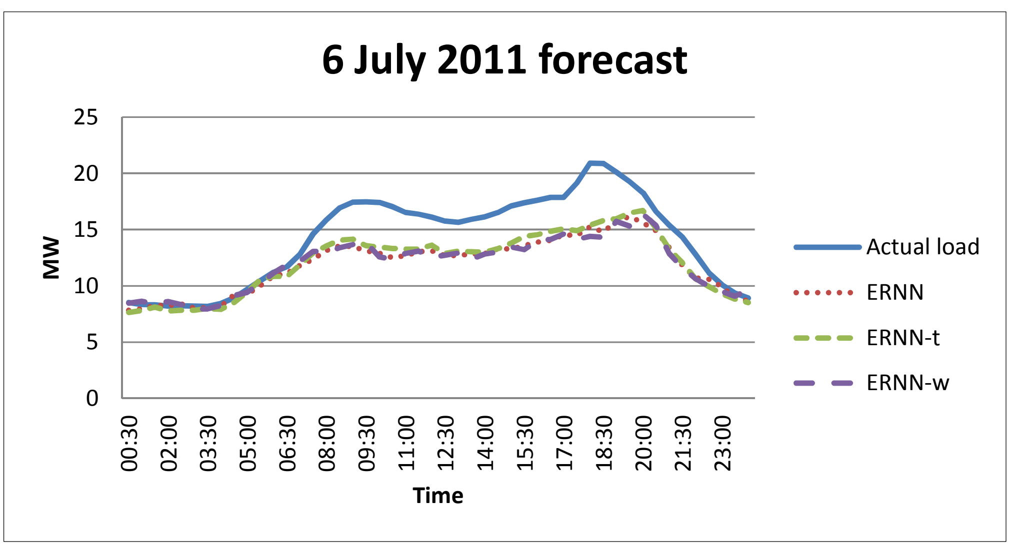


Figure 3.23: Wednesday July 2011 ERNN load forecast

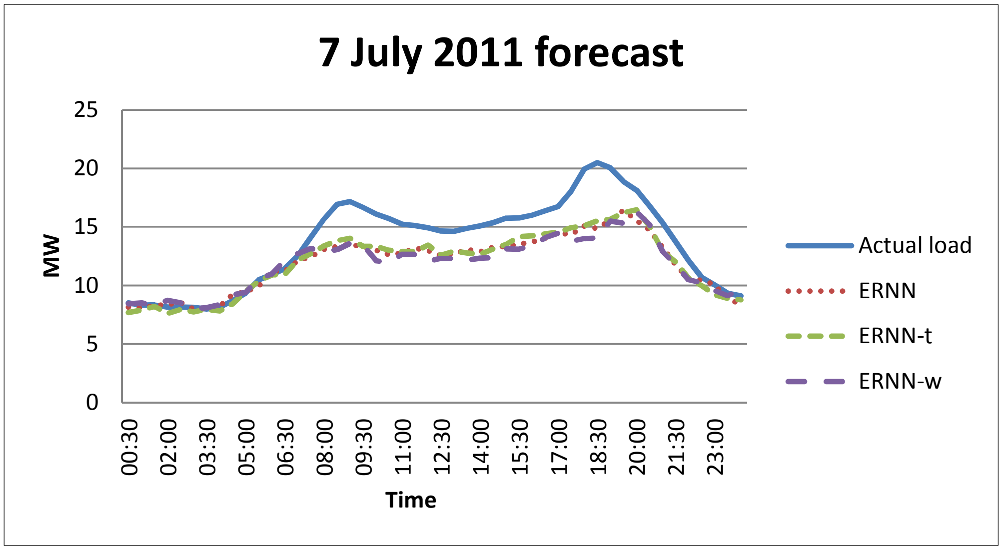


Figure 3.24: Thursday July 2011 ERNN load forecast

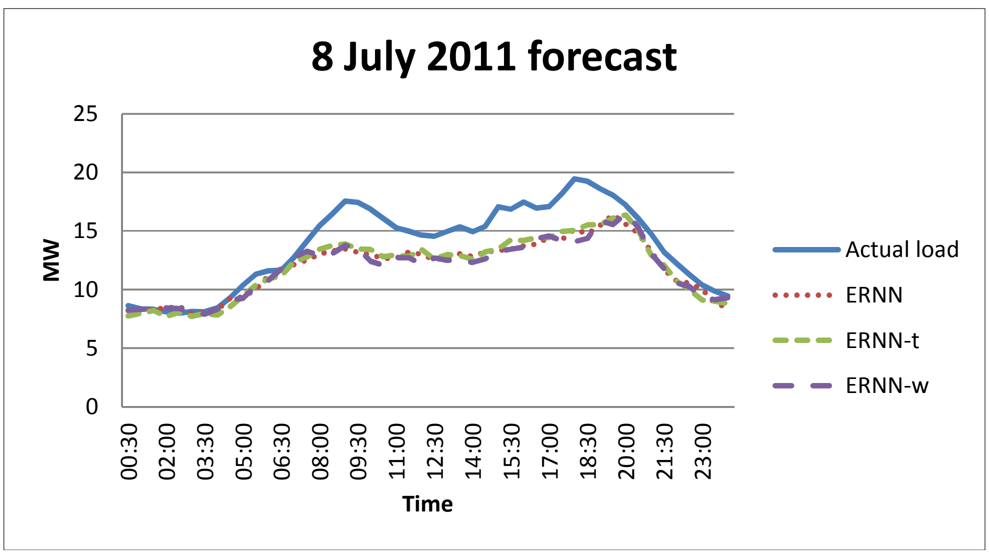


Figure 3.25: Friday July 2011 ERNN load forecast

Table 3.14: ERNN MAPE for weekdays in July 2011

Day	ERNN	ERNN-t	ERNN-w
Wednesday 6 July 2011	14.01	13.12	14.17
Thursday 7 July 2011	11.24	10.83	12.41
Friday 8 July 2011	12.04	11.75	12.29

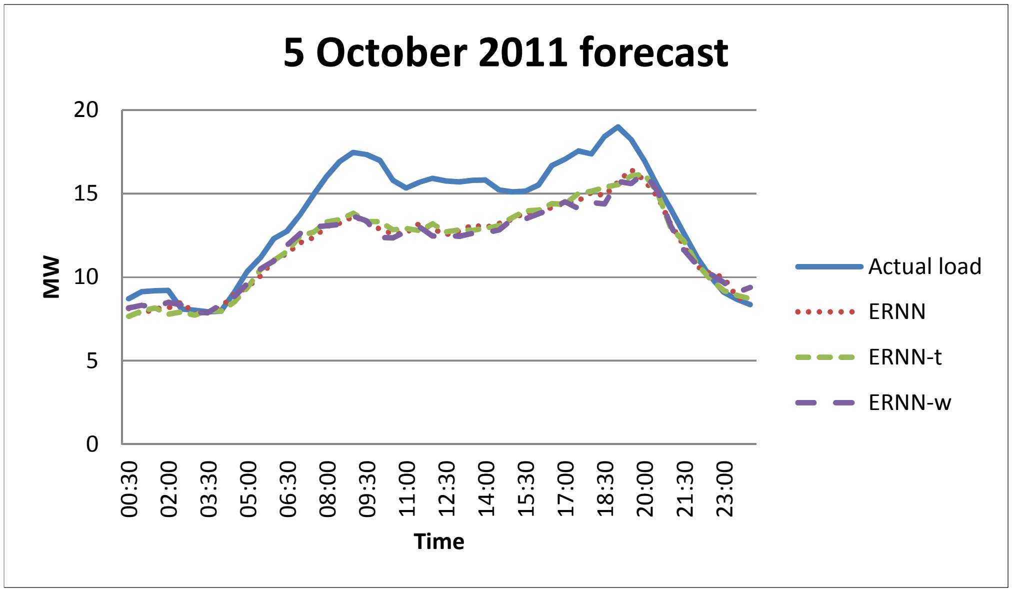


Figure 3.26: Wednesday October 2011 ERNN load forecast

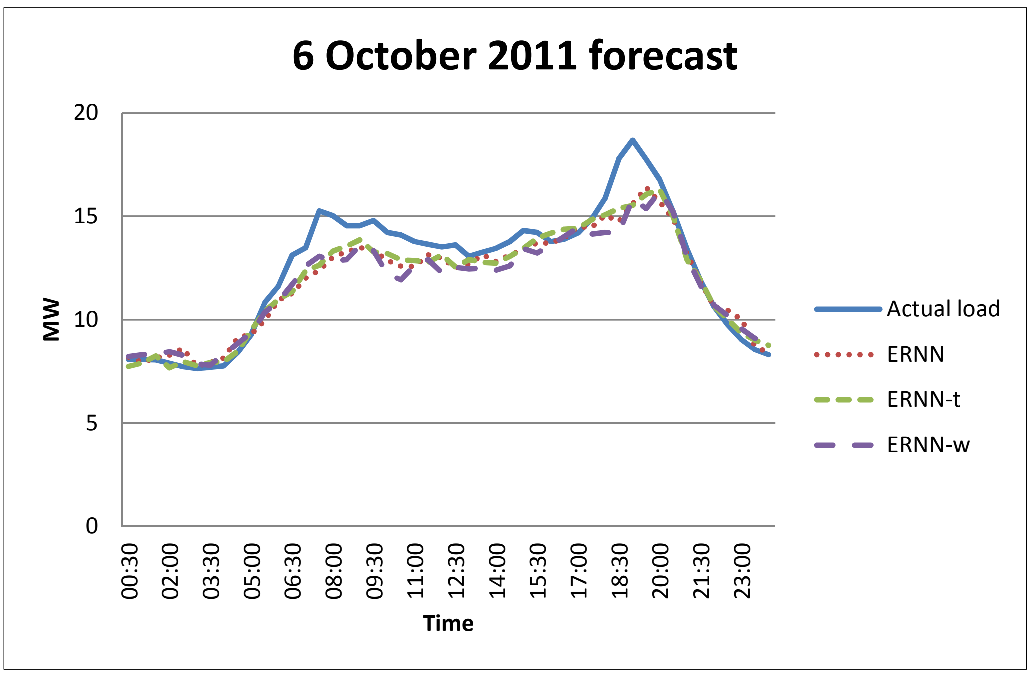


Figure 3.27: Thursday October 2011 ERNN load forecast

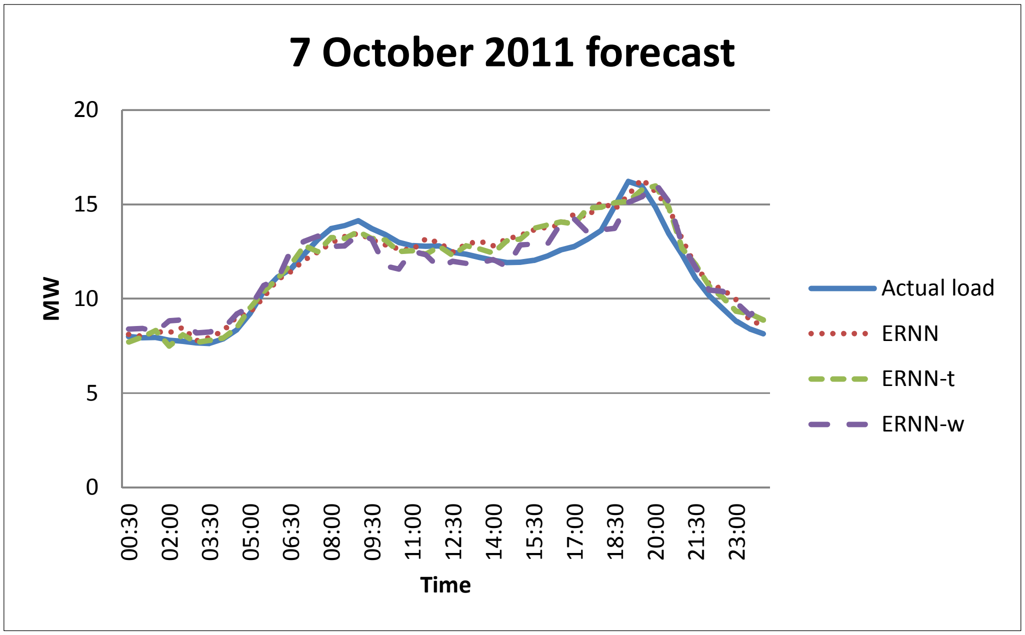


Figure 3.28: Friday October 2011 ERNN load forecast

Table 3.15: ERNN MAPE for weekdays in October 2011

Day	ERNN	ERNN-t	ERNN-w
Wednesday 5 October 2011	12.01	11.46	12.27
Thursday 6 October 2011	6.00	5.16	6.94
Friday 7 October 2011	5.67	4.96	6.63

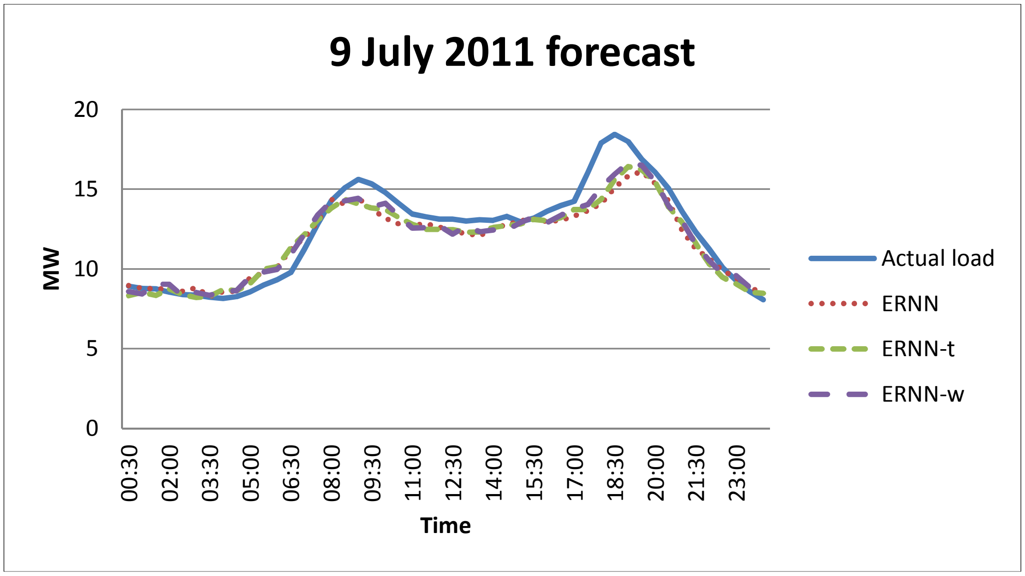


Figure 3.29: Saturday July 2011 ERNN load forecast

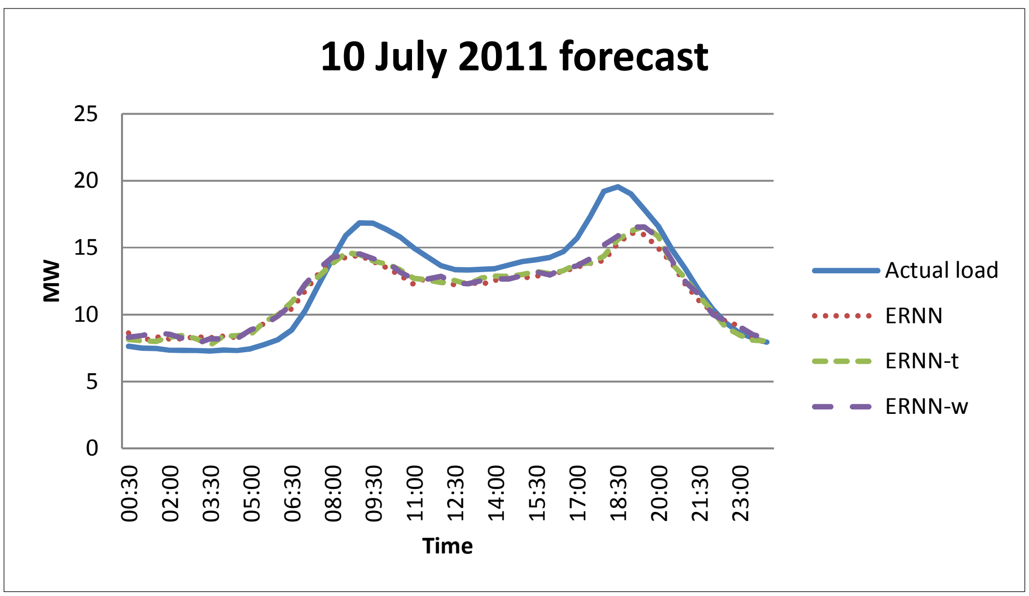


Figure 3.30: Sunday July 2011 ERNN load forecast

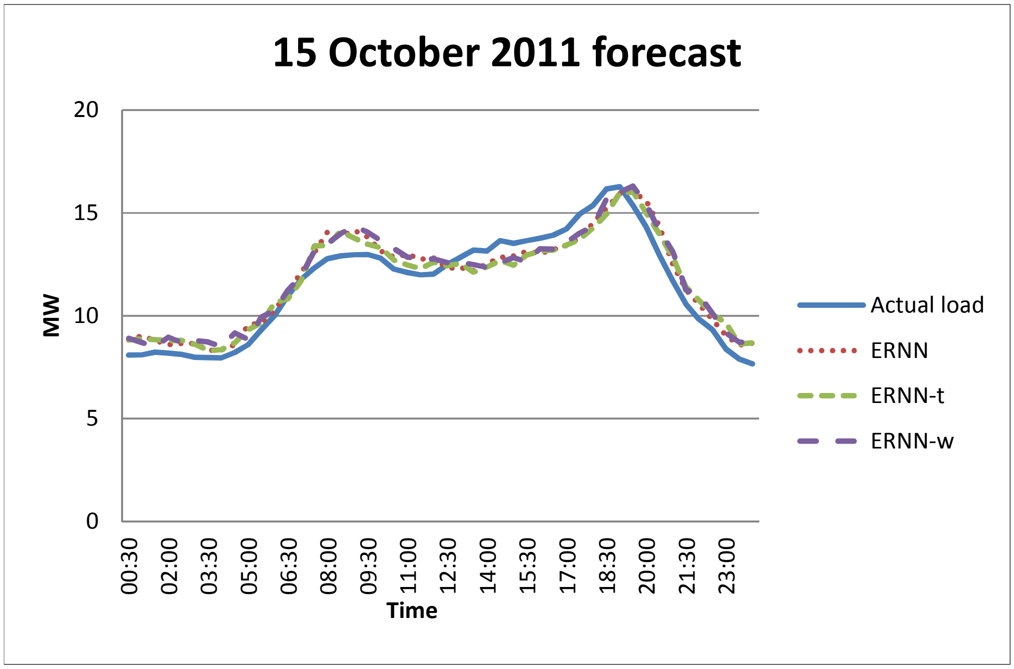


Figure 3.31: Saturday October 2011 ERNN load forecast

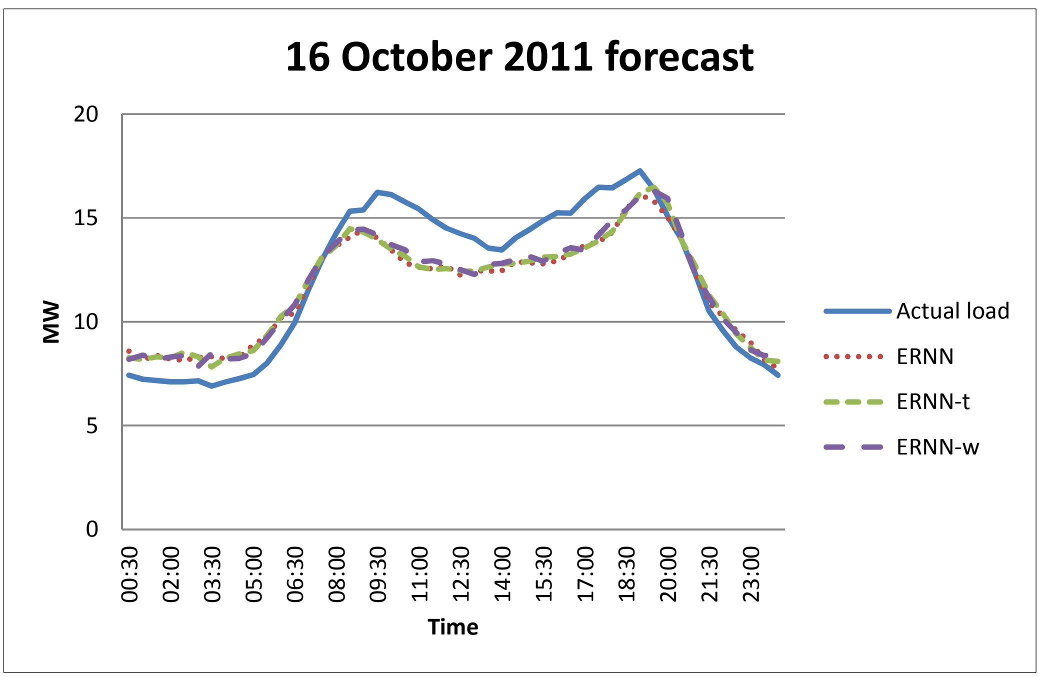


Figure 3.32: Sunday October 2011 ERNN load forecast

Table 3.16: ERNN MAPE for weekends in 2011

Day	ERNN	ERNN-t	ERNN-w
Saturday 9 July 2011	6.05	5.92	5.52
Sunday 10 July	11.47	10.47	10.82
Saturday 15 October 2011	6.23	6.31	6.78
Sunday 16 October 2011	10.72	10.66	10.16

It can be seen from the figures that the ERNN networks attempt to follow the actual load profiles for the forecast days however there are errors in excess of 10% in most of the forecasts. The network with the best performance based on MAPE is found to be the temperature sensitive ERNN-t with a minimum forecasting error of 4.96% and a maximum forecasting error of 13.12% (Table3.17). The average forecasting errors obtained for ERNN, ERNN-t and ERNN-w were 9.46%, 876% and 9.16% respectively.

These high forecasting errors were not expected by the author. It was assumed that based on literature [25, 53], the forecasting errors would be considerably better than those obtained using ANN. Marin *et al*. in [53] developed a model that classified each day according to its load profile by means of self-organising feature maps and built and trained recurrent neural networks for each class. In this study, the data was only classified as weekend and weekday loads. It is the author's assumption that the ERNN requires individually defined day types and the training done according to that day type. It is possible that the ERNN was unable to extrapolate a clear relationship of day types and loads with only a weekend and weekday load classification.

In [25], the authors forecasted loads one hour ahead and utilised the previous hours forecast as input. It is the author's assumption that the inclusion of the previous hours load forecast may have an influence in minimising forecasting errors. In this research, a full day's half-hourly load forecast was conducted. It is possible that these high forecasting errors obtained by ERNN in this study may be improved by training and forecasting for specific day types as conducted in [53]. The results shown in Table 3.17 follow the same performance trend as the ANN networks in that the temperature sensitive ERNN performed better in comparison to the other two networks.

Table 3.17: MAPE Performance evaluation for ERNN networks

Day	ERNN	ERNN-t	ERNN-w
Wednesday 9 March 2011	9.22	7.82	7.48
Thursday 10 March 2011	9.12	7.56	6.54
Friday 11 March 2011	9.25	7.83	7.10
Wednesday 6 July 2011	14.01	13.12	14.17
Thursday 7 July 2011	11.24	10.83	12.41
Friday 8 July 2011	12.04	11.75	12.29
Saturday 9 July 2011	6.05	5.92	5.52
Sunday 10 July	11.47	10.47	10.82
Wednesday 5 October 2011	12.01	11.46	12.27
Thursday 6 October 2011	6.00	5.16	6.94
Friday 7 October 2011	5.67	4.96	6.63
Saturday 15 October 2011	6.23	6.31	6.78
Sunday 16 October 2011	10.72	10.66	10.16
Average MAPE	9.46	8.76	9.16

For comparative purposes, a statistical load forecasting method namely Multiple Linear Regression (MLR) was tested using Microsoft Excel (XLSTAT). This is discussed in the next section.

3.4 Multiple linear regression method

As discussed in section 2.2.1, traditional methods have been applied to short term load forecasting. Multiple linear regression method had been utilised in this study for comparative purposes. MLR was chosen as it is a simple method that is able to take multiple variables as input. The inputs to the study are hourly temperature, hourly humidity, day of the week and hourly previous day load data. The Regression analysis tool performs linear regression analysis by using the "least squares" method to fit a line through a set of observations. XLSTAT was used to conduct the study. Table 3.18 shows the result of the regression study obtained from XLSTAT. The values

Table 3.18: MLR regression parameters

Source	Value	Standard error	t	Pr > |t|	Lower bound (95%)	Upper bound (95%)
Intercept	3.019	0.130	23.172	< 0.0001	2.764	3.275
input load	0.832	0.006	148.074	< 0.0001	0.821	0.843
Weekday	-0.230	0.008	-29.612	< 0.0001	-0.245	-0.214
Temp	0.025	0.003	8.456	< 0.0001	0.019	0.031
humidity	-0.009	0.001	-11.017	< 0.0001	-0.011	-0.007

The Values in Table 3.18 are then used to create the straight line equation as follows:

$$y = 3.019 + 0.832 * input\ load - 0.230 * weekday + 0.025 * Temperature - 0.009 * humidity \quad (3.9)$$

Equation (3.9) was then used to conduct the prediction for select days in the year 2011. The results are shown in the next section and are compared with the results of the ANN networks.

3.4.1 Simulation results

The following Figures 3.33 to 3.45 illustrate the predictions obtained by using MLR. These are compared on the same axes with the ANN forecasting results.

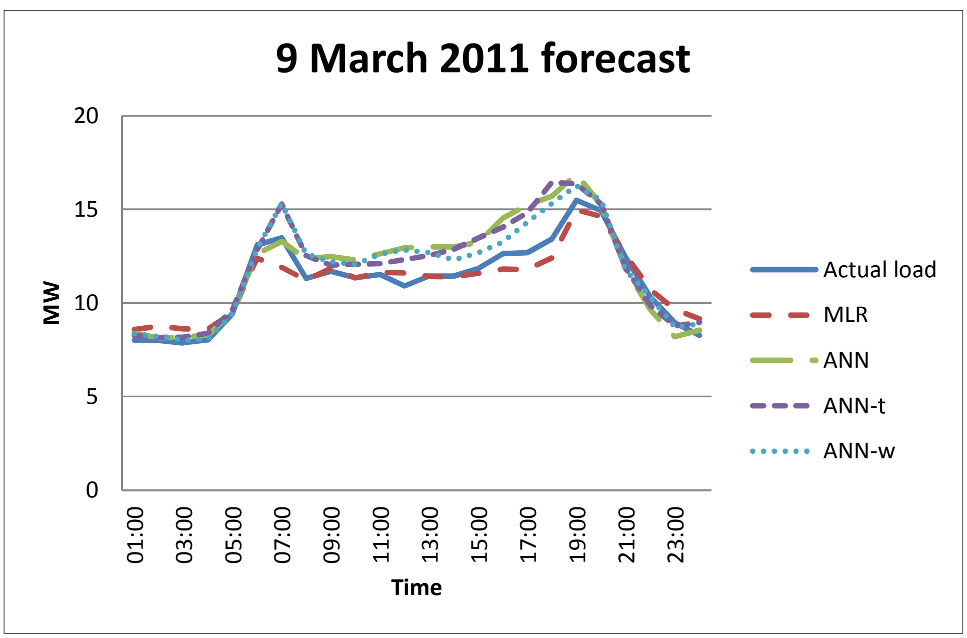


Figure 3.33: Wednesday March 2011 MLR vs. ANN

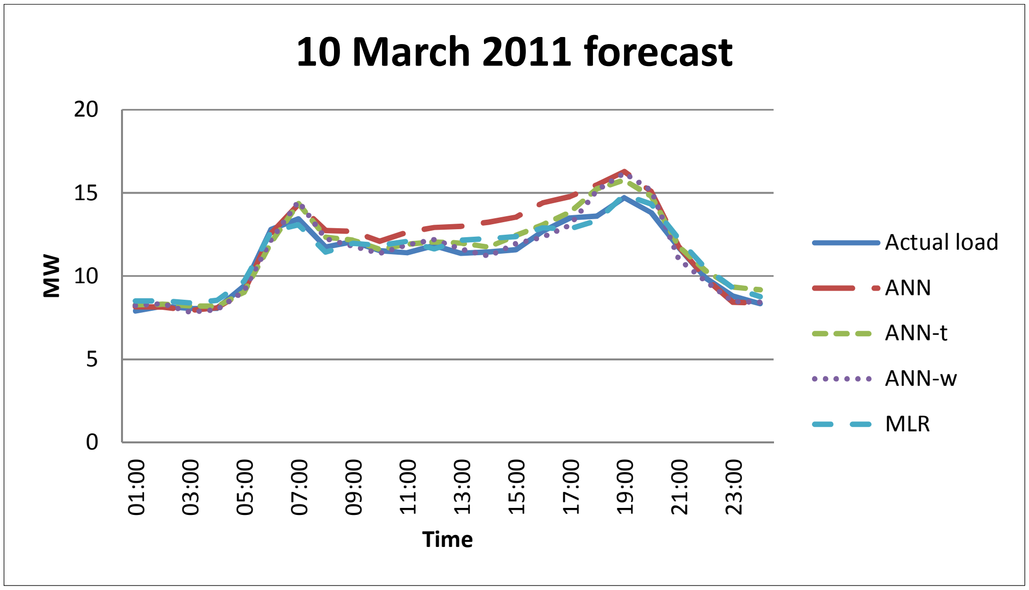


Figure 3.34: Thursday March 2011 MLR vs. ANN

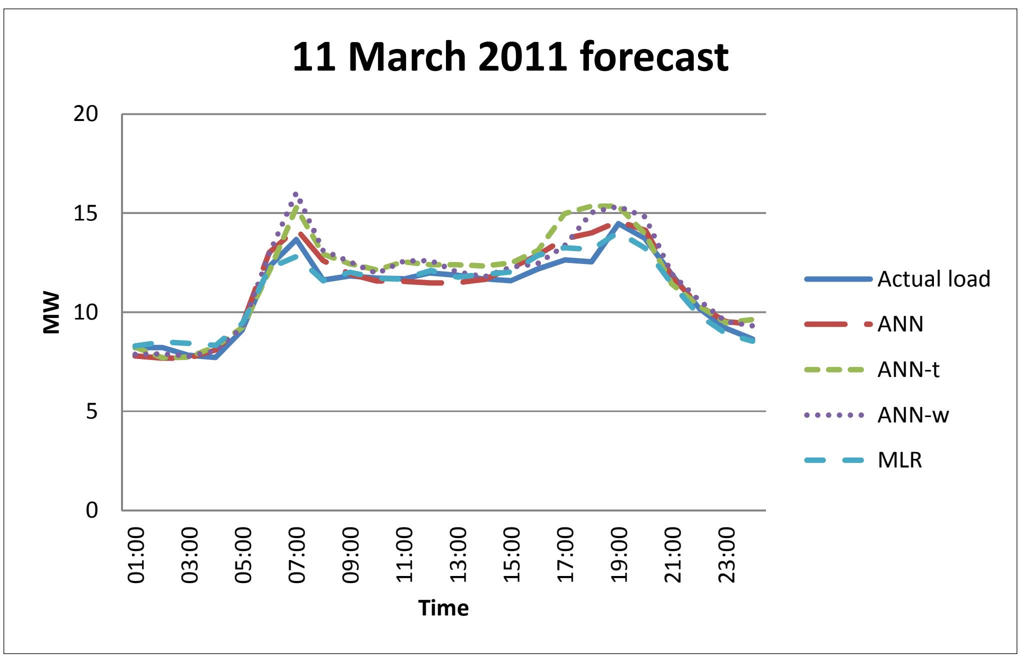


Figure 3.35: Friday March 2011 MLR vs. ANN

Table 3.19: MLR vs ANN March MAPE

Day	ANN	ANN-t	ANN-w	MLR
Wednesday 9 March	8.05	7.46	6.35	4.7
Thursday 10 March	6.91	4.40	3.93	4.1
Friday 11 March	4.32	6.13	5.54	3.1

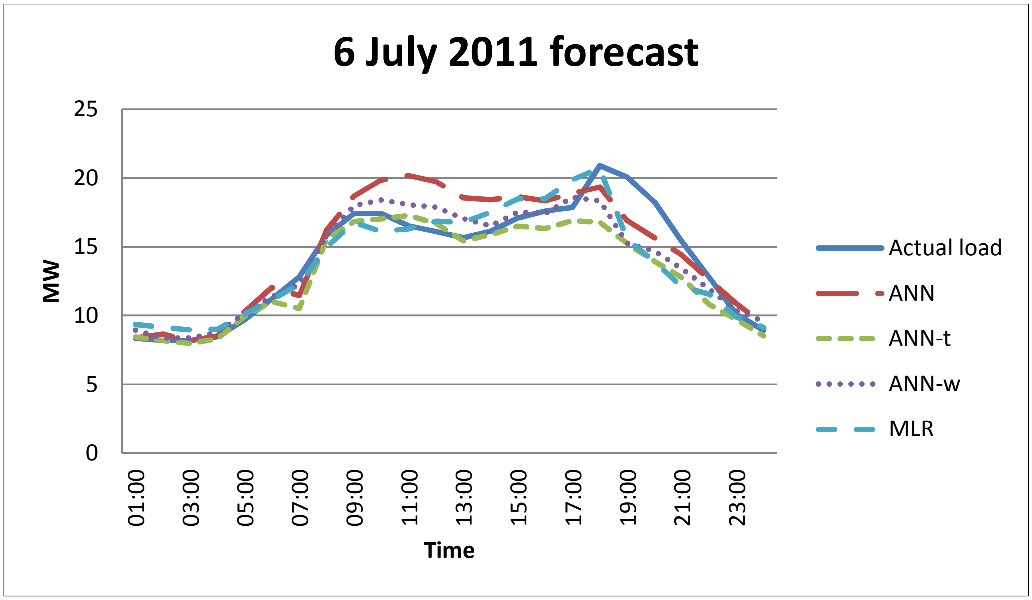


Figure 3.36: Wednesday July 2011 MLR vs. ANN

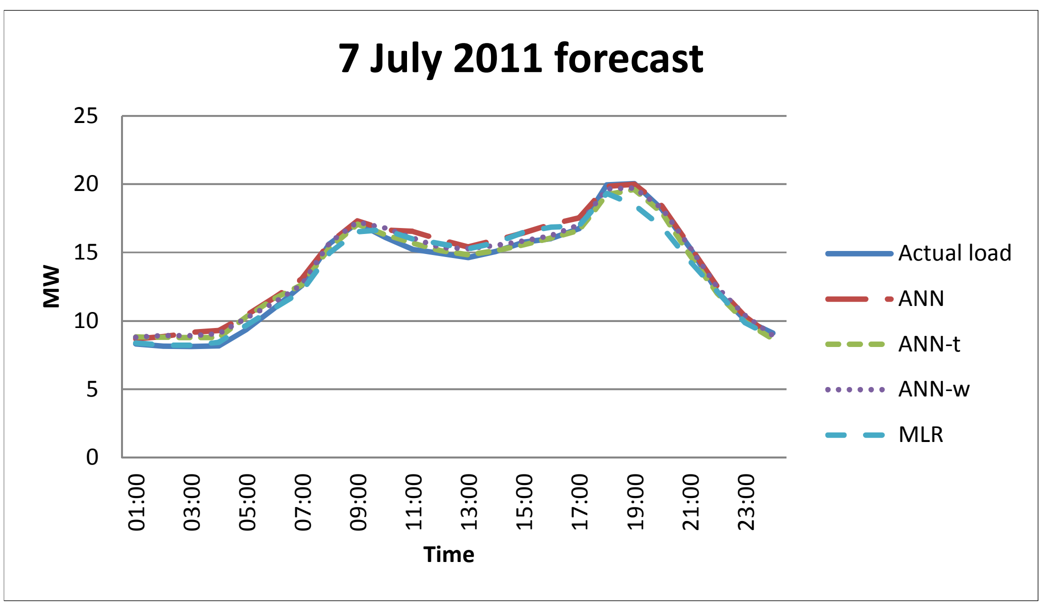


Figure 3.37: Thursday July 2011 MLR vs. ANN

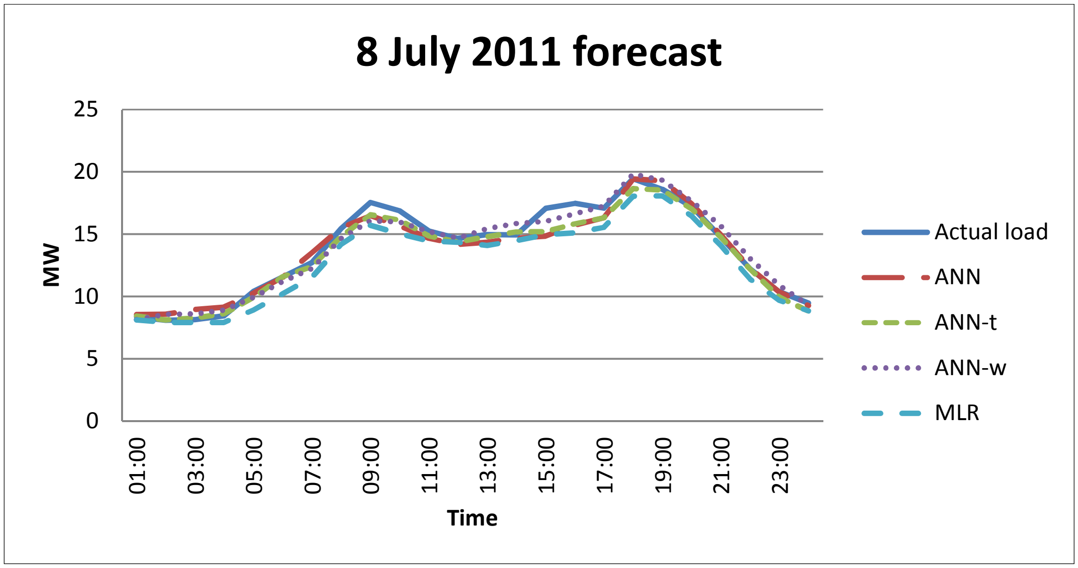


Figure 3.38: Friday July 2011 MLR vs. ANN

Table 3.20: July MLR vs ANN MAPE

Day	ANN	ANN-t	ANN-w	MLR
Wednesday 6 July	9.52	7.18	6.69	8.1
Thursday 7 July	4.88	3.05	3.55	3.4
Friday 8 July	4.07	3.18	4.13	7.0

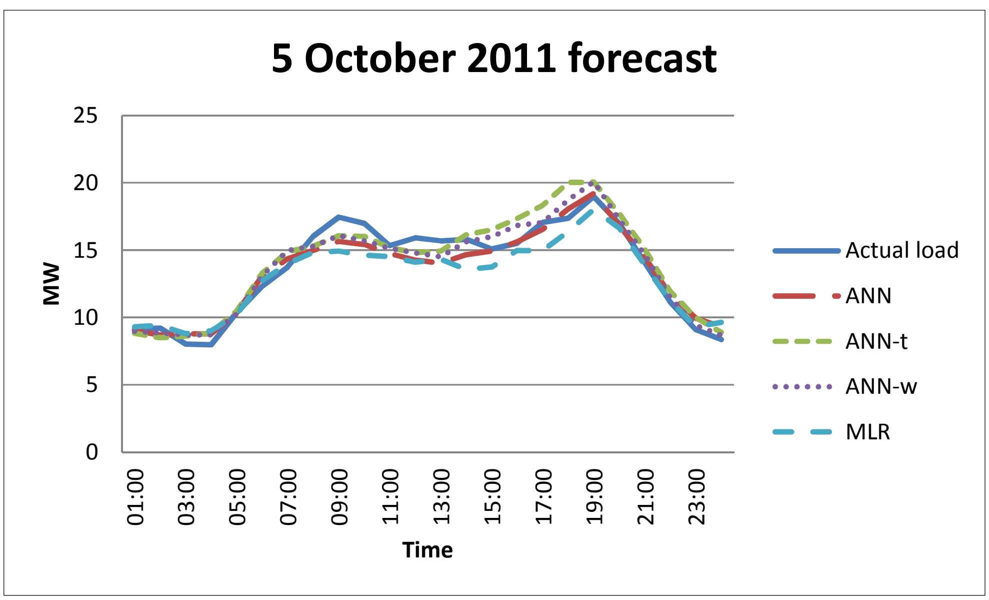


Figure 3.39: Wednesday October 2011 MLR vs. ANN

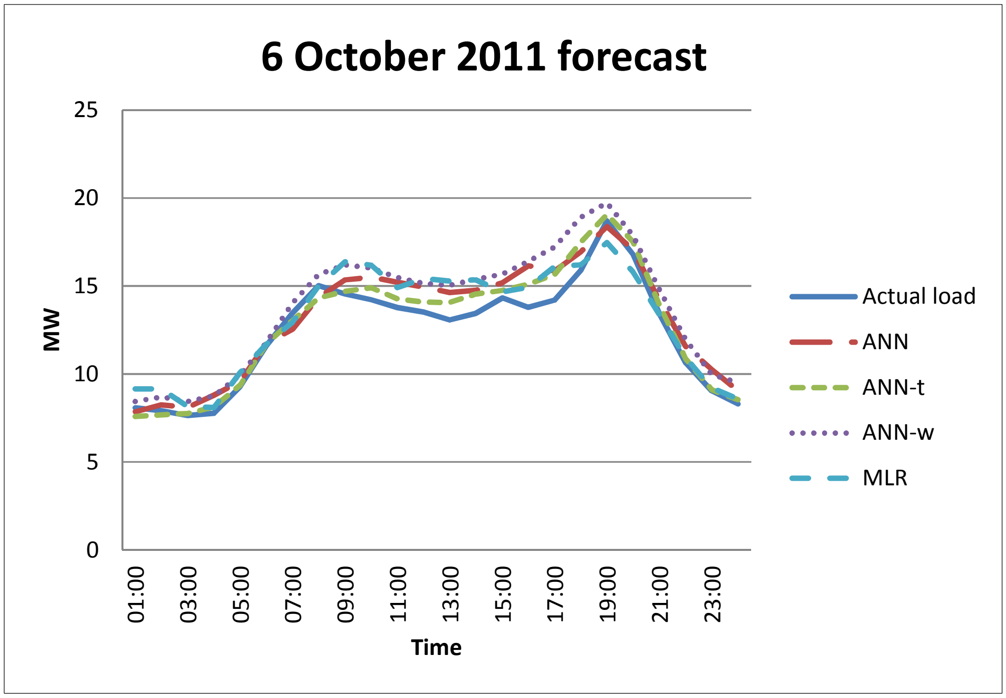


Figure 3.40: Thursday October 2011 MLR vs. ANN

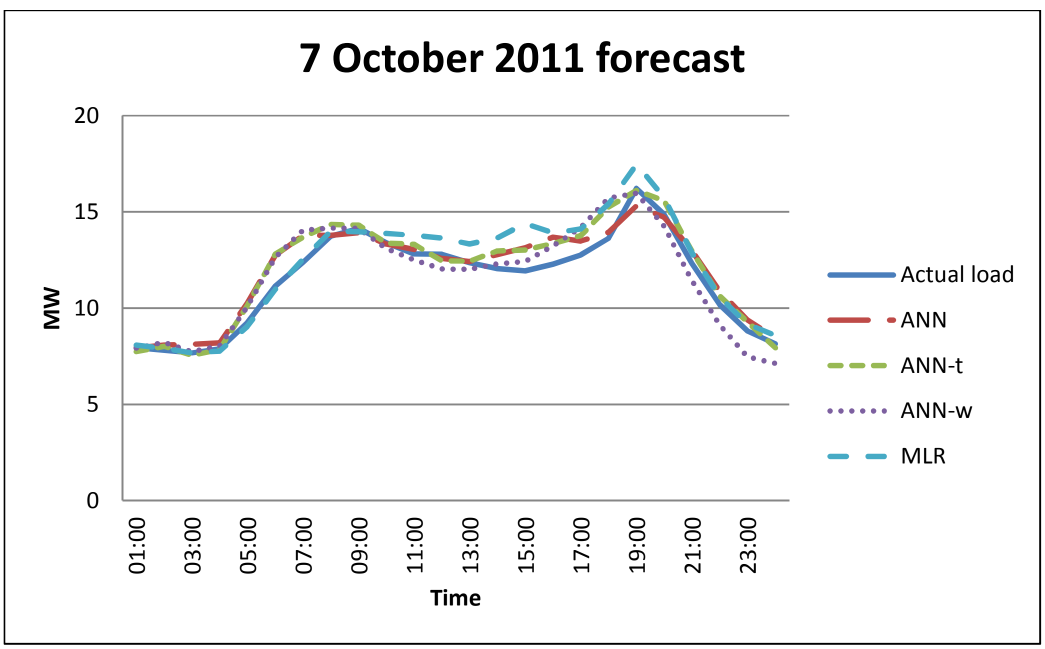


Figure 3.41: Friday October 2011 MLR vs. ANN

Table 3.21: October MLR vs ANN MAPE

Day	ANN	ANN-t	ANN-w	MLR
Wednesday 5 October	5.77	6.59	5.08	6.9
Thursday 6 October	7.55	4.09	10.33	7.5
Friday 7 October	5.01	4.95	6.65	5.9

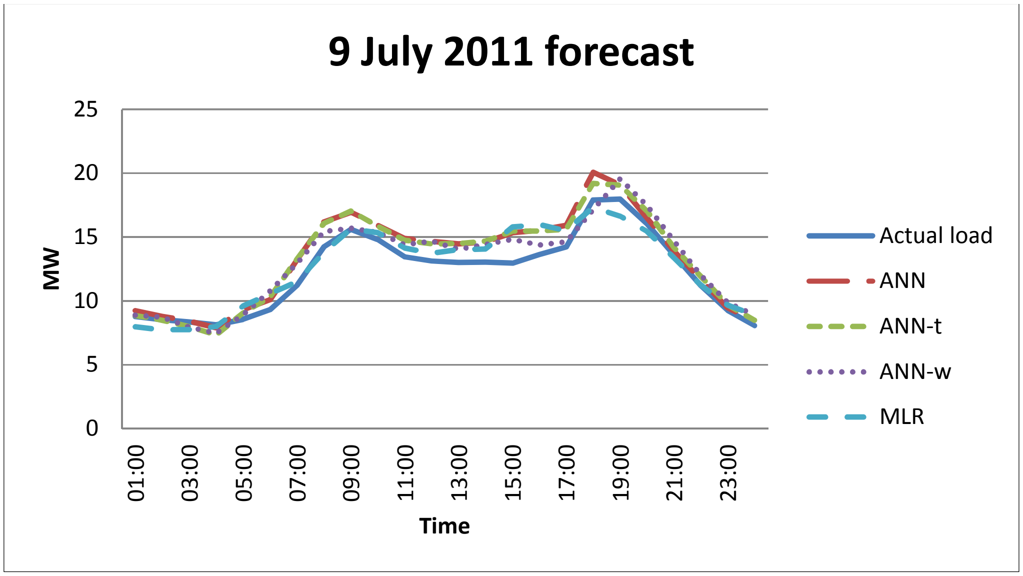


Figure 3.42: Saturday July 2011 MLR vs. ANN

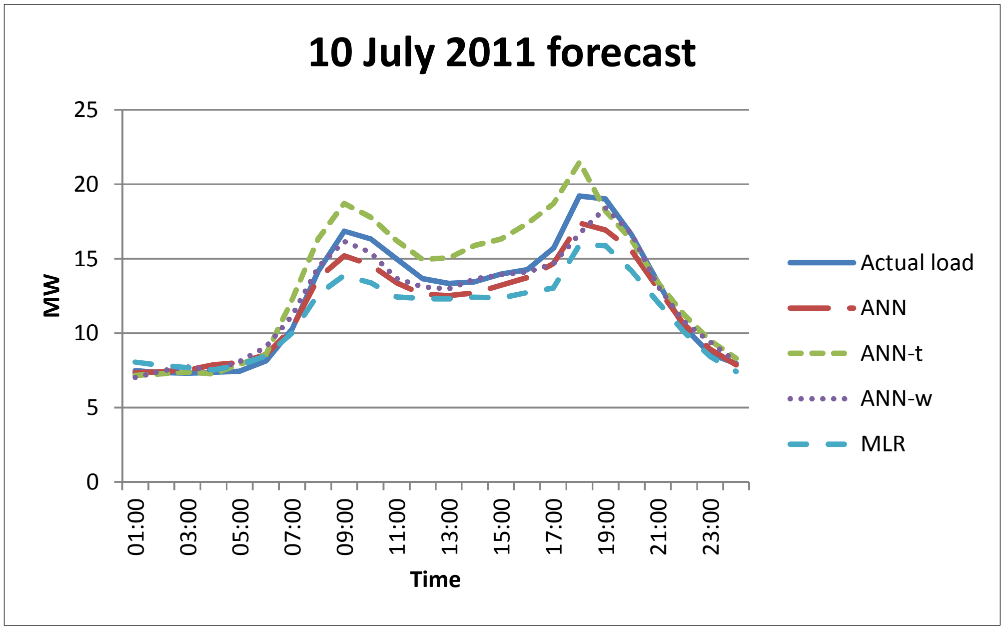


Figure 3.43: Sunday July 2011 MLR vs. ANN

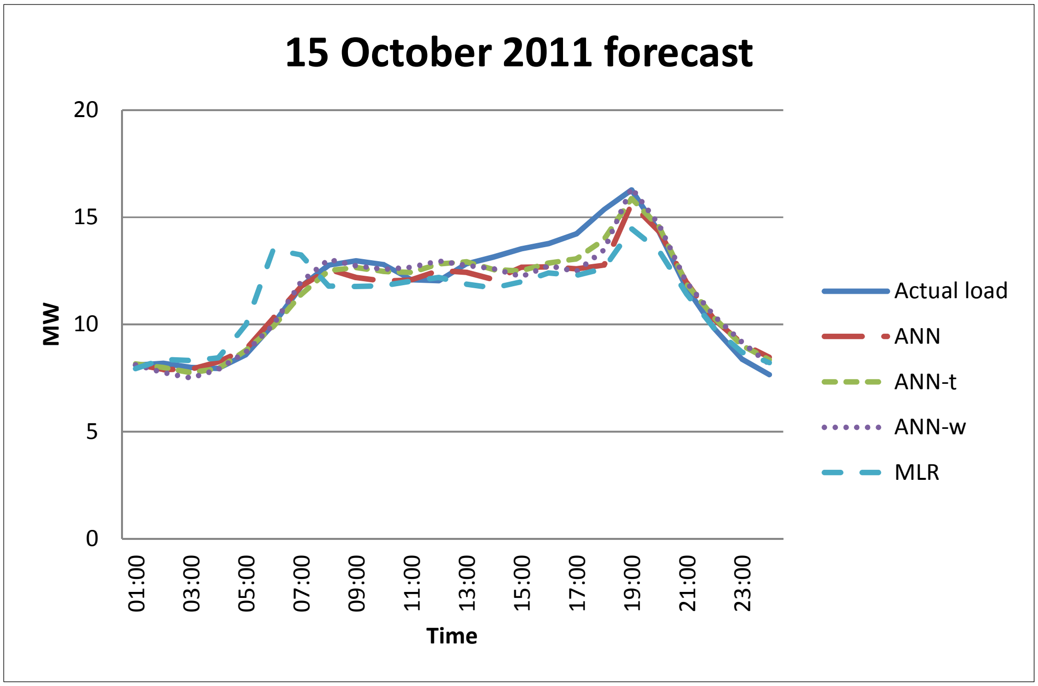


Figure 3.44: Saturday October 2011 MLR vs. ANN

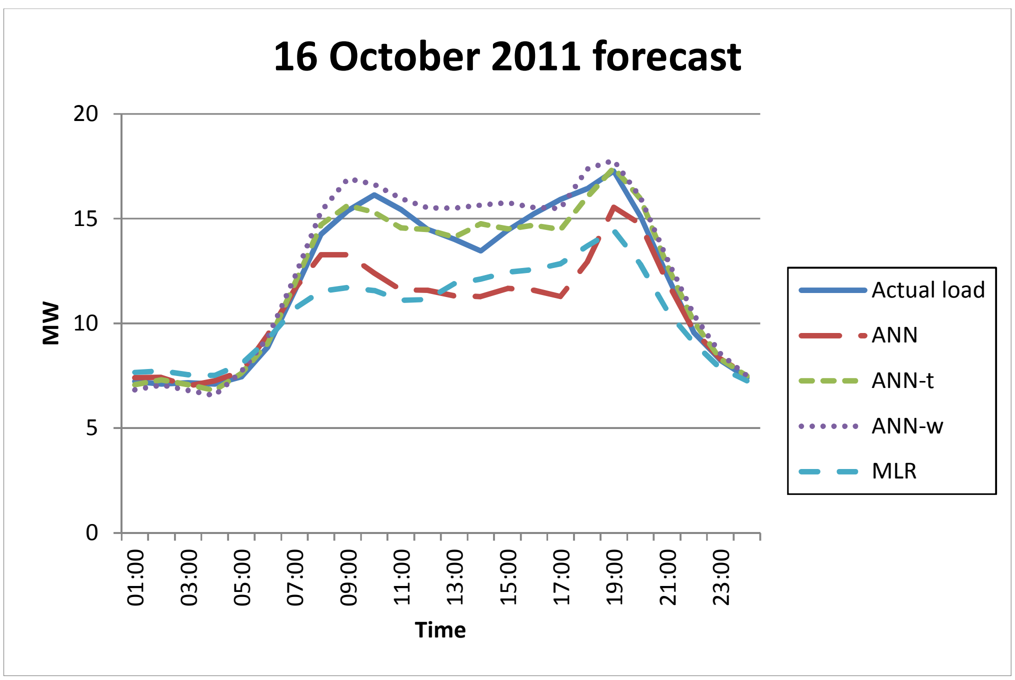


Figure 3.45: Sunday October 2011 MLR vs. ANN

Table 3.22: MLR vs ANN weekend MAPE

Day	ANN	ANN-t	ANN-w	MLR
Saturday 9 July	8.08	8.27	7.12	6.9
Sunday 10 July	5.63	9.16	4.98	9.7
Saturday 15 October	4.75	3.58	4.59	8.6
Sunday 16 October	10.96	3.47	5.64	13.3

3.5 Discussion of results

From the simulation results of each of the tested neural networks, one can see that the performance of the ERNN models was not satisfactory in this study. Performance errors in excess of 10% were encountered. The load forecasts presented by these networks, although they drew a similar shape, were consistently lower than the actual load hence the large forecasting errors. MLR produced fairly good forecasts with a minimum error of 3.1% and a maximum of 13.3% and an average error of 6.86%. It was also able to follow the behaviour of the load profile fairly well however, in most instances it failed to achieve a good forecast of the peak hour load. The ANN networks, particularly ANN-t (temperature sensitive ANN), were able to predict this fairly accurately. Table 3.23 shows the performances of all tested networks and it is evident that ANN-t is the best performing network.

These results thus strengthen the hypothesis that the addition of load affecting variables such as temperature and humidity has a positive effect on the accuracy of load forecasting. However not all variables can be added, it is recommended that a correlation study of these variables with the load needs to be conducted first. In this study, one can deduce from the MAPE in Table 3.23 that temperature is the most important factor in a load forecast. In an effort to improve the forecasting errors produced by the neural networks, a hybrid method consisting of the best performing network i.e. ANN-t and Particle Swarm Optimisation (PSO) is investigated in the next chapter. Although there are a number of other CI techniques available, PSO was selected because of its simplicity and ease of use as compared to GAs. As discussed in chapter 2, GA's require a number of variables such as mutation, reproduction and crossover factors etc. to be manipulated in order to get an optimal solution whereas PSO only requires selection of a few variables such as the particle swarm size.

Table 3.23: Performance evaluation for all tested neural networks and MLR

Day	ANN	ANN-t	ANN-w	ERNN	ERNN-t	ERNN-w	MLR
Wednesday 9 March	8.05	7.46	6.35	9.22	7.82	7.48	4.7
Thursday 10 March	6.91	4.40	3.93	9.12	7.56	6.54	4.1
Friday 11 March	4.32	6.13	5.54	9.25	7.83	7.10	3.1
Wednesday 6 July	9.52	7.18	6.69	14.01	13.12	14.17	8.1
Thursday 7 July	4.88	3.05	3.55	11.24	10.83	12.41	3.4
Friday 8 July	4.07	3.18	4.13	12.04	11.75	12.29	7.0
Saturday 9 July	8.08	8.27	7.12	6.05	5.92	5.52	6.9
Sunday 10 July	5.63	9.16	4.98	11.47	10.47	10.82	9.7
Wednesday 5 October	5.77	6.59	5.08	12.01	11.46	12.27	6.9
Thursday 6 October	7.55	4.09	10.33	6.00	5.16	6.94	7.5
Friday 7 October	5.01	4.95	6.65	5.67	4.96	6.63	5.9
Saturday 15 October	4.75	3.58	4.59	6.23	6.31	6.78	8.6
Sunday 16 October	10.96	3.47	5.64	10.72	10.66	10.16	13.3
Average MAPE	6.58	5.5	5.74	9.46	8.76	9.16	6.86

Chapter 4 : Short Term Load Forecasting using Particle Swarm Optimisation (PSO)

This section discusses the use of a particle swarm optimisation method in combination with an ANN model in order to improve the forecasting error.

4.1 Overview of Particle Swarm Optimisation

Particle swarm optimisation (PSO) is an evolutionary computation technique discovered by Eberhart and Kennedy [40]. It is a population based search procedure where individuals (particles) change their position with time [12]. The model has a set of n particles each representing a dimension of solution space. These particles move in the solution space in order to obtain the optimal solution. Each particle changes its position based on the influence by its nearest neighbour and tries to imitate the best solution. The particles position changes based on the velocity it has over a certain number of iterations. While the particles search for the best position, they influence each other and thereafter converge to an optimal solution. This algorithm has been used to train neural networks to solve a number of problems such as described by Kennedy *et al.* [55].

The basic principle is described as follows [11]:

1. A particle i is associated with a current position in the search space $\mathrm{w_i}$, a current velocity $\mathrm{v_i}$ and a personal best position $\mathrm{p_i}$. A swarm s consists of particles i.
2. The personal best position $\mathrm{p_i}$ corresponds to the particles position in the solution space where particle i presents the smallest error as determined by an objective function f.
3. The global best position p_g represents the position with the lowest error amongst all the $\mathrm{p_i}$'s

The personal and global best positions are updated according the equations (4.1) and (4.2)

$$p_i(t+1) = \begin{cases} p_i(t), & if\ f(p_i(t)) \leq f(w_i(t+1)) \\ w_i(t+1), & if\ f(p_i(t)) > \ f(w_i(t+1)) \end{cases} \tag{4.1}$$

$$p_g \in \ \{p_0(t), p_1(t), \dots, p_s(t)\} \text{ and } p_g = min\{f\big(p_0(t)\big), f(p_1(t)), \dots f(p_s(t))\} \tag{4.2}$$

Each particles velocity and position is updated using equations (4.3) and (4.5) as follows:

$$v_{i,j}(t+1) = \omega v_{i,j}(t) + c_1 r_{1,i}(t)\left(p_{i,j}(t) - w_{i,j}(t)\right) + c_2 r_{2,i}(t)(p_{g(i,j)}(t) - w_{i,j}(t)) \tag{4.3}$$

where :

r_1 and r_2 are random values between 0 and 1 and are used to affect the stochastic nature of the algorithm

$w_{i,j}$, $p_{i,j}$ and $v_{i,j}$ are the current position, current personal best position and velocity of the j^{th} dimension of the i^{th} particle

c_1 and c_2 are the acceleration coefficients which control how far a particle can move in a single iteration. These are typically set to the value of 2 however they can be varied and range between 0 and 4

ω denotes the inertia weight which is used to control the convergence of the PSO and is calculated as in (4.4)

$$\omega = \omega_{max} - \frac{\omega_{max} - \omega_{min}}{maxit} * iter \tag{4.4}$$

where:

ω_{max} and ω_{min} are the maximum and minimum inertia weight

$maxit$ is the maximum number of iterations

$iter$ denotes the current iteration value

The new velocity is then added to the current position of the particle as follows in order to get its next position:

$$w_i(t+1) = w_i(t) + v_i(t+1) \tag{4.5}$$

The acceleration coefficients are typically set to the value of 2 however they can be varied and range between 0 and 4 [41].

The fitness of the i^{th} particle is measured by the optimisation function f which is configured according to the problem that needs to be solved. The particle with the minimum error is chosen as the best particle. In this PSO method the objective is to obtain the particle with the lowest Mean square error (MSE) obtained from a neural network.

MSE is calculated as follows:

$$f(w_i) = \frac{1}{N}\sum_{k=1}^{N}\left[\frac{1}{O}\sum_{l=1}^{O}\{T_{kl} - P_{kl}(w_i)\}^2\right] \tag{4.6}$$

where:

f is the fitness value

T_{kl} is the target output value

P_{kl} is the predicted output value based on the position vector w_i

N is the number of training set samples

O is the number of output neurons

The particle with the lowest fitness obtained in equation (4.6) is then utilised in the ANN to forecast the next day's half hourly load. The procedure is discussed in the next section.

4.2 PSO – ANN Applied to Short Term Forecasting Model

In this research, PSO instead of BP is used to train the ANN network by altering the weights such that the resulting MSE for the training data is reduced. This process runs until a stop criterion is met which in this case is until the maximum number of iterations has been reached (see Figure 4.1). The values in Table 4.1 were selected based on the guideline provided by Hu in [41] and validated by experimentation with other values.. The particle position in the search space of the PSO corresponds to the weights of the ANN. The fitness function *f* corresponds to the *MSE* of the ANN network. Each particle represents a possible solution of weights. The ANN topology is taken from ANN-t which is the best performing neural network that was discussed in chapter 3 and the PSO variables are shown in Table 4.1. ANN-t is optimised by PSO in order to further minimise the forecasting errors.

Table 4.1: PSO Variables

PSO variables	Values
coefficient (c1)	2
coefficient (c2)	2
inertia (w)	0.4<= w <= 1.0
number of particles (S)	20
Max iterations	100
constriction factor(K)	0.729
r1	random (0,1)
r2	random (0,1)
Vmax	-0.4 <= Vmax <= 0.4

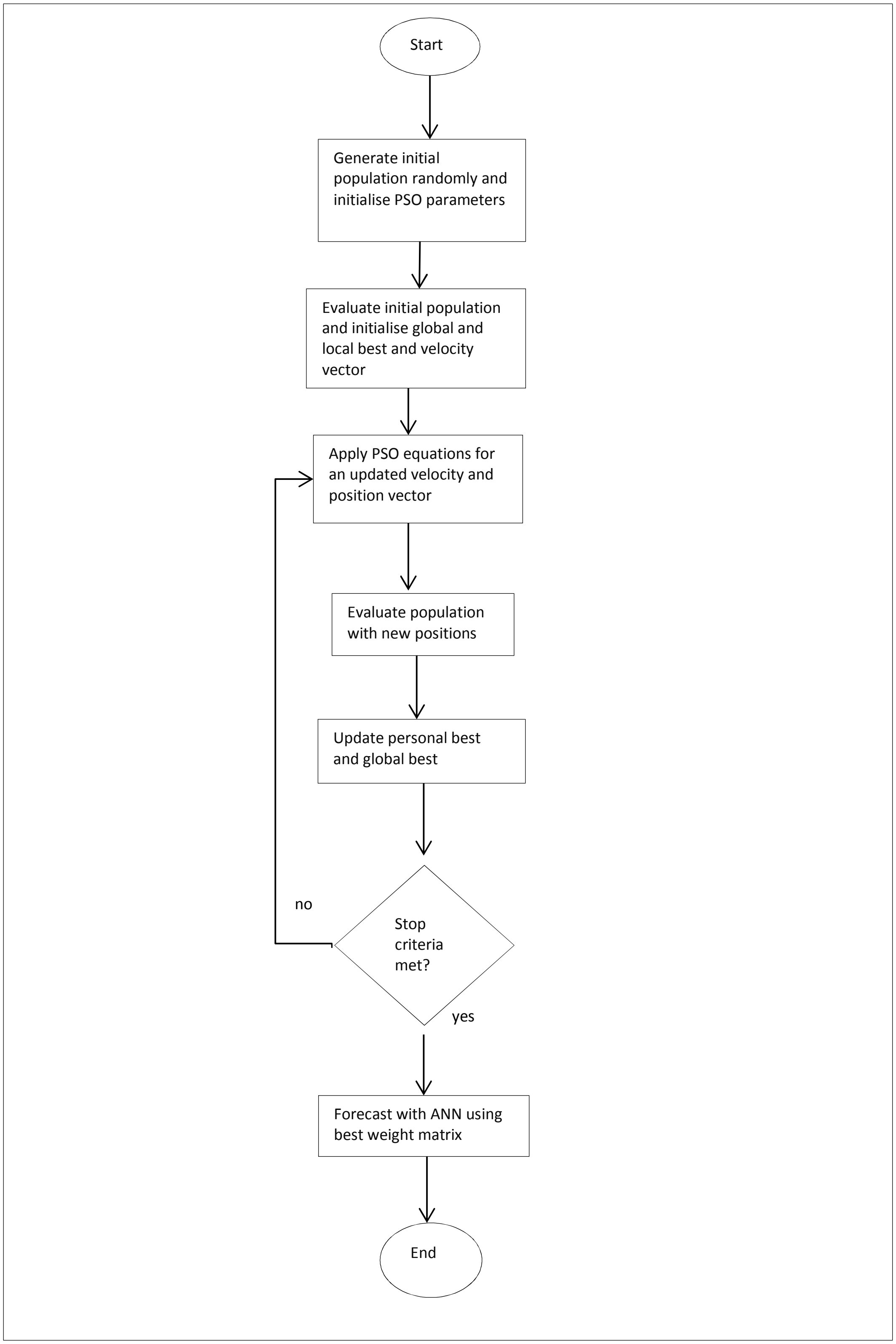


Figure 4.1: PSO- ANN process

4.3 Simulation Results

The following Figures 4.2 to 4.14 illustrate the predicted loads in the year 2011 with the results of ANN-t placed on the same set of axes.

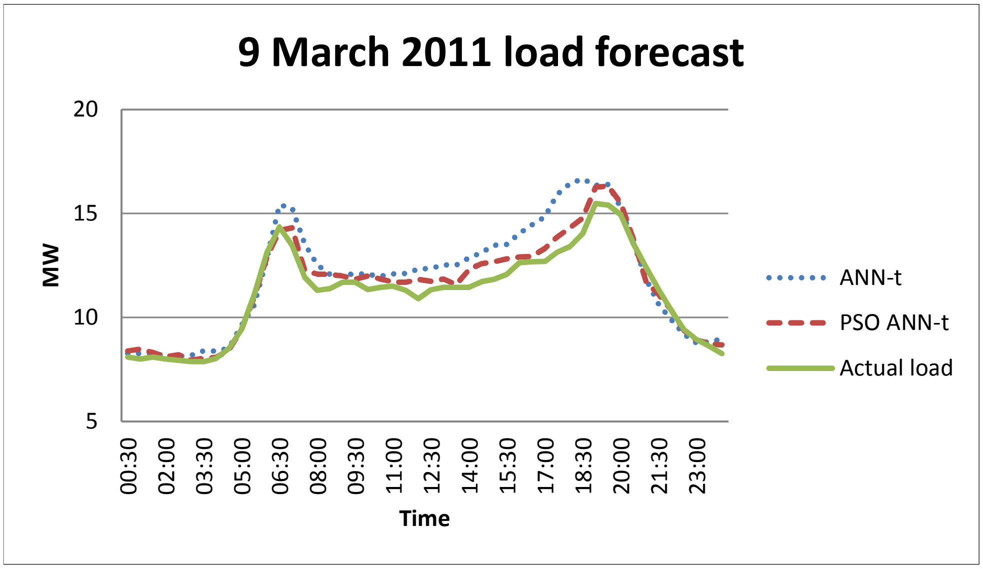


Figure 4.2: Wednesday March 2011 PSO ANN

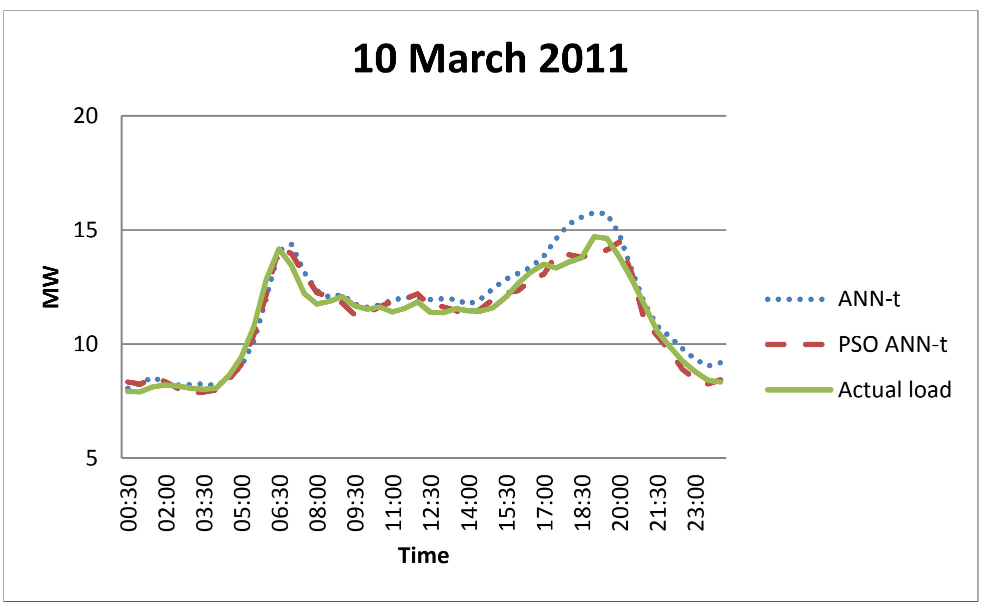


Figure 4.3: Thursday March 2011 PSO ANN

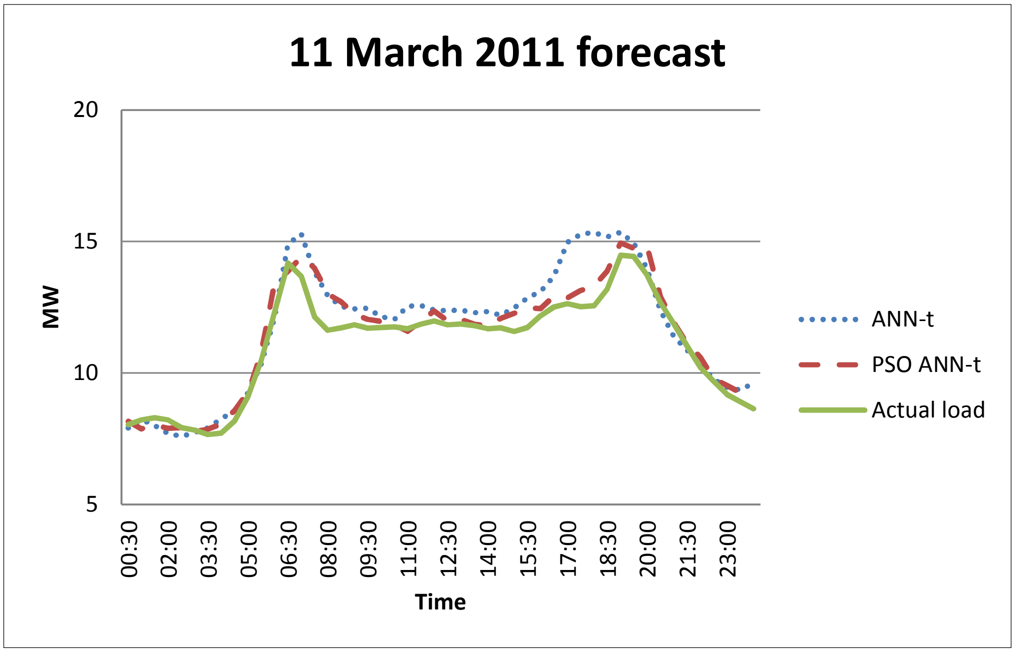


Figure 4.4: Friday March 2011 PSO ANN

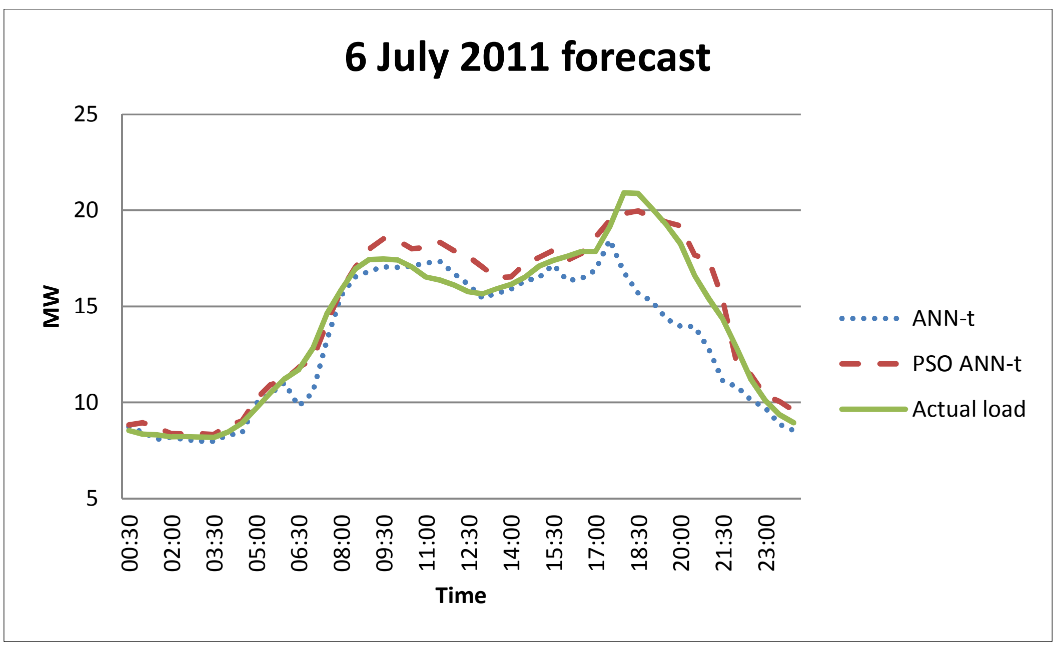


Figure 4.5: Wednesday July 2011 PSO ANN

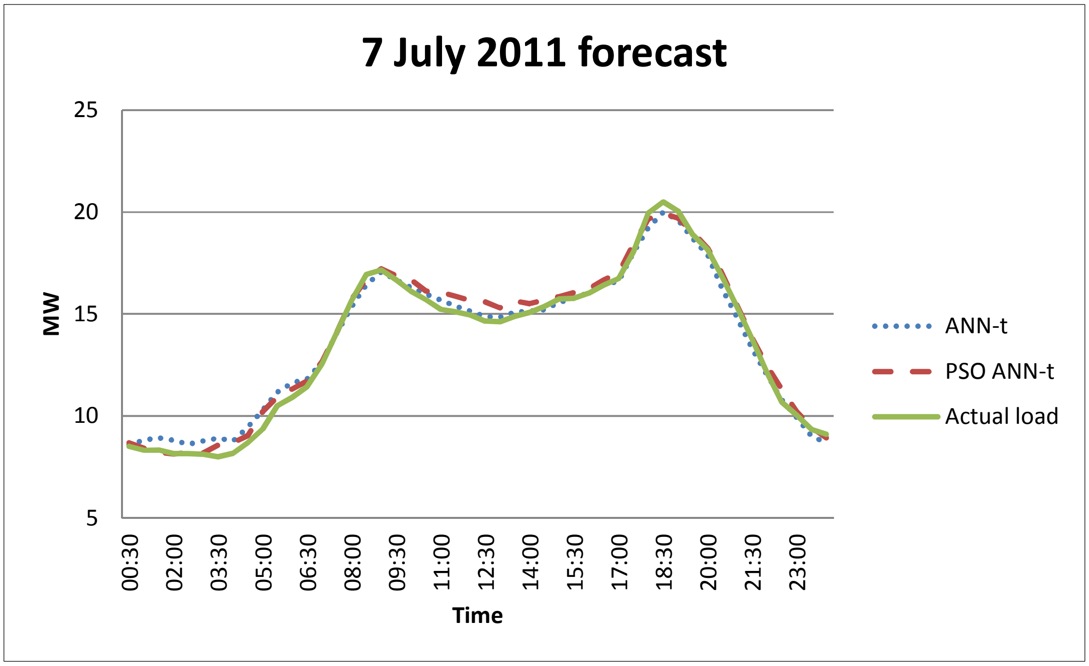


Figure 4.6: Thursday July 2011 PSO ANN

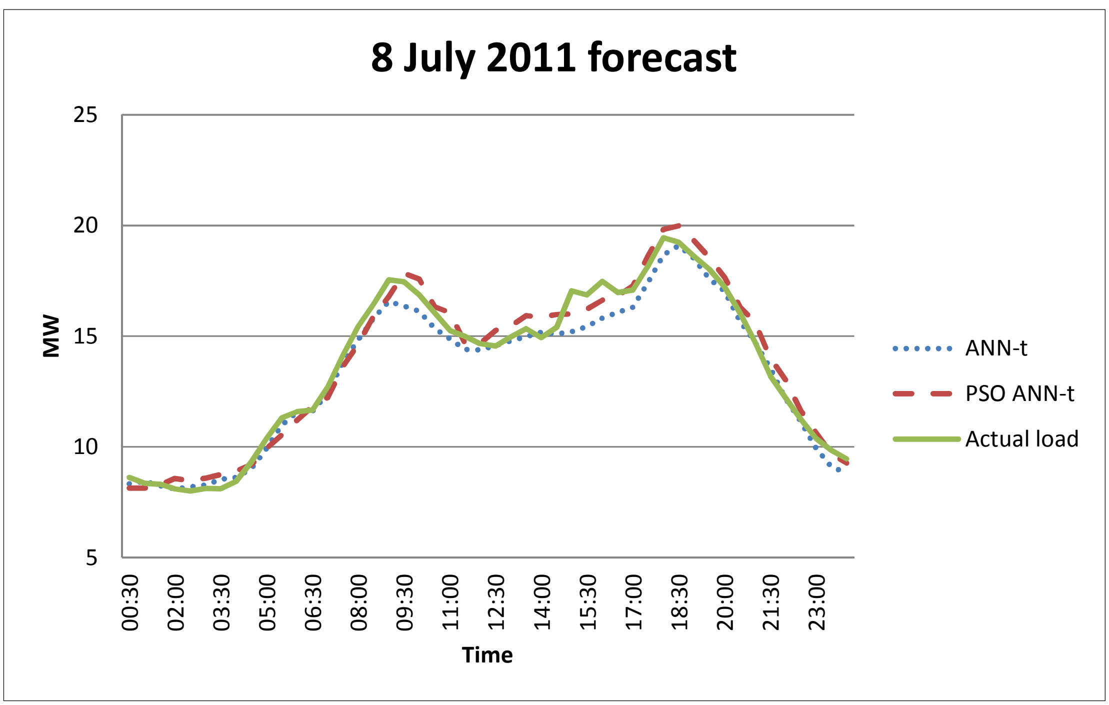


Figure 4.7: Friday July 2011 PSO ANN

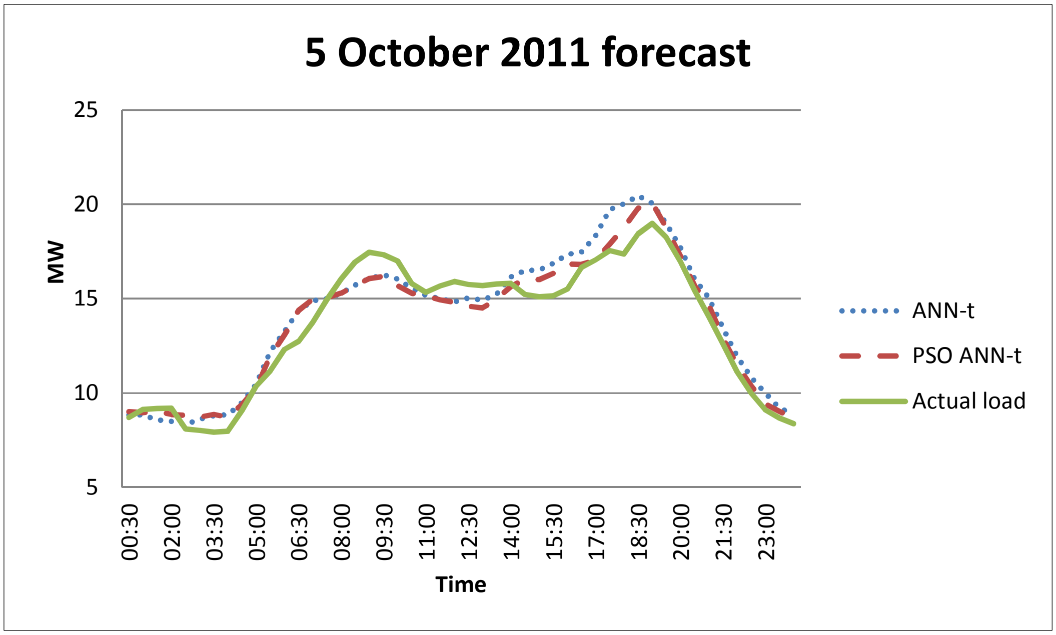


Figure 4.8: Wednesday October 2011 PSO ANN

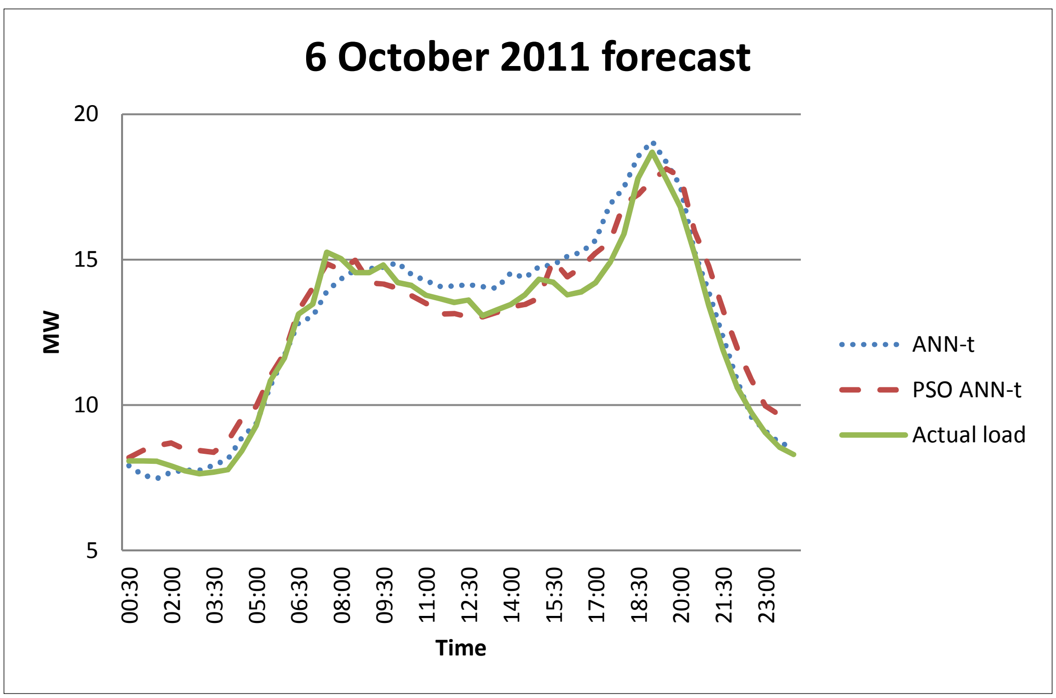


Figure 4.9: Thursday October 2011 PSO ANN

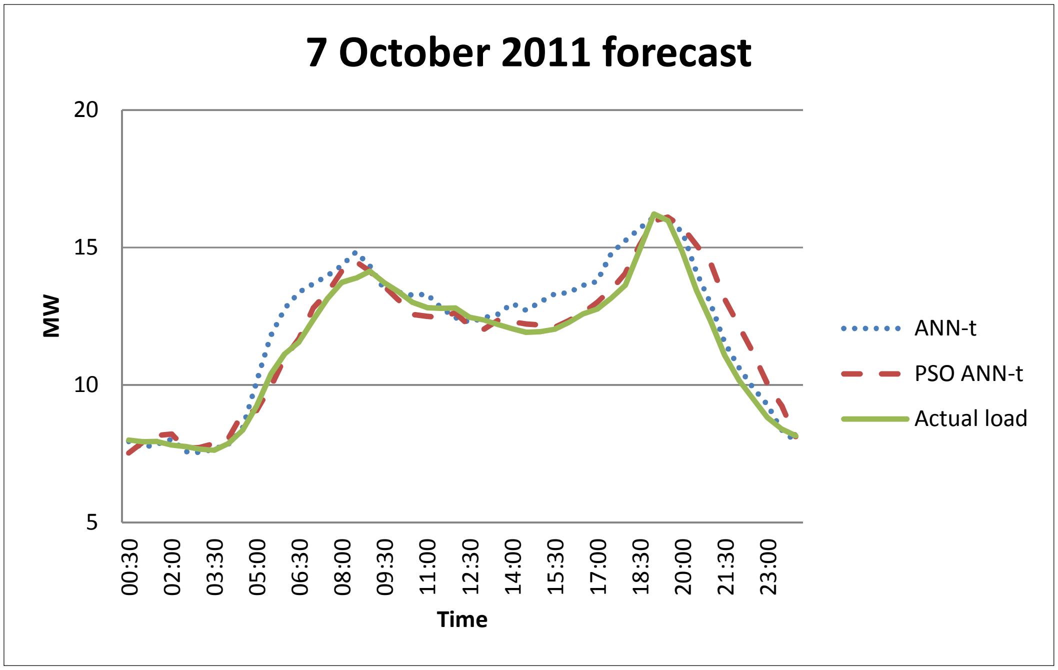


Figure 4.10: Friday October 2011 PSO ANN

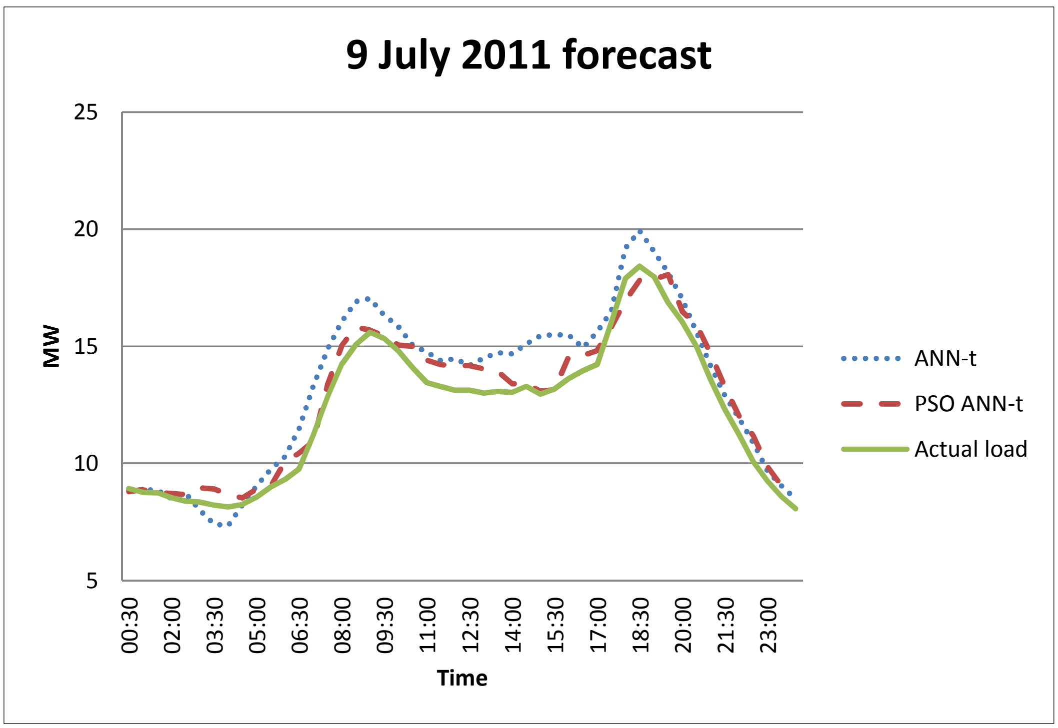


Figure 4.11: Saturday July 2011 PSO ANN

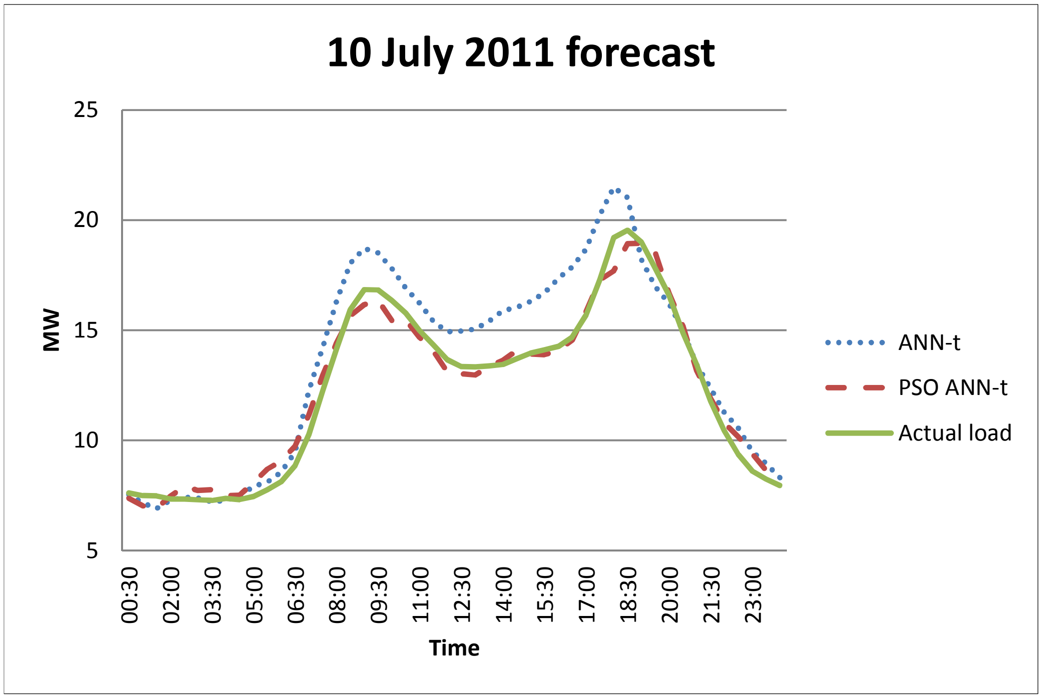


Figure 4.12: Sunday July 2011 PSO ANN

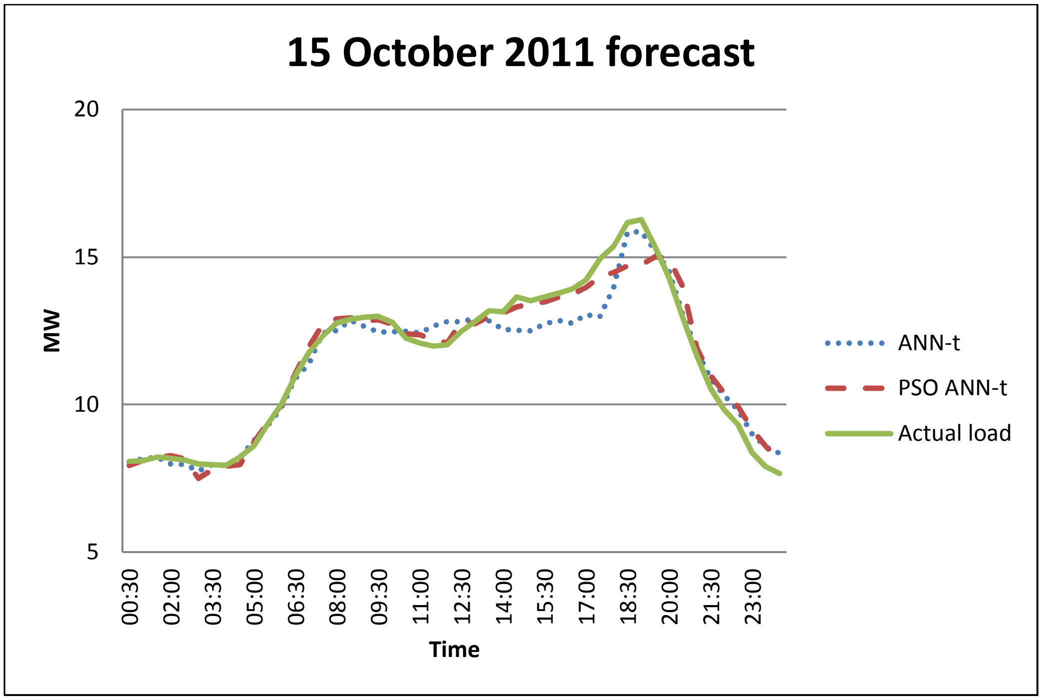


Figure 4.13: Saturday October 2011 PSO ANN

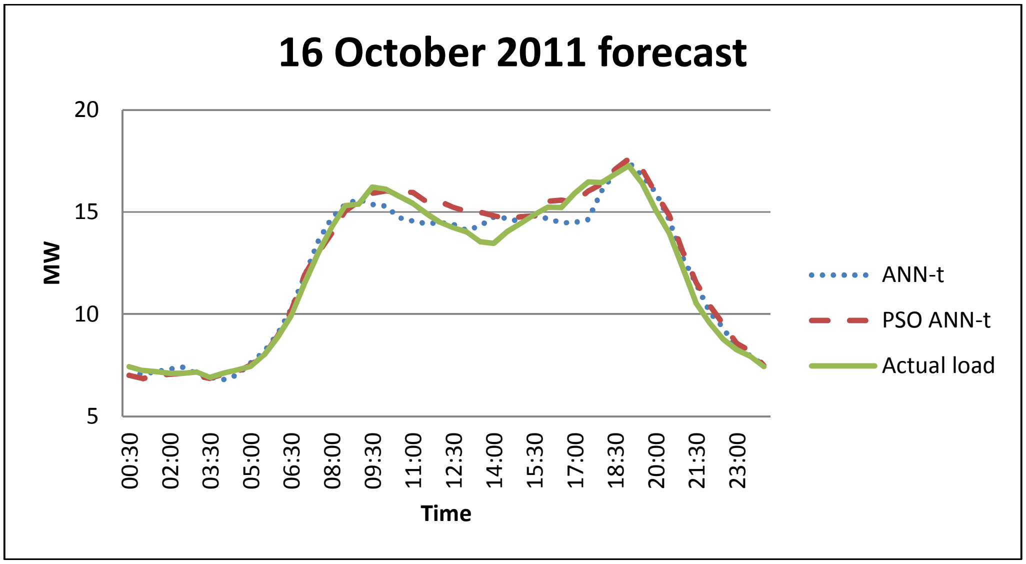


Figure 4.14: Sunday October 2011 PSO ANN

Table 4.2: Performance comparison of ANN-t and PSO ANN-t

Day	ANN-t	PSO-ANN-t
Wednesday 9 March 2011	7.46	3.52
Thursday 10 March 2011	4.40	2.82
Friday 11 March 2011	6.13	3.66
Wednesday 6 July 2011	7.18	4.47
Thursday 7 July 2011	3.05	2.51
Friday 8 July 2011	3.18	3.71
Saturday 9 July 2011	8.27	4.76
Sunday 10 July	9.16	3.95
Wednesday 5 October 2011	6.59	5.08
Thursday 6 October 2011	4.09	5.70
Friday 7 October 2011	4.95	4.22
Saturday 15 October 2011	3.58	2.77
Sunday 16 October 2011	3.47	3.39

From Figures 4.2 to 4.14, one can see that the PSO ANN-t produces results that show a great improvement in forecasting errors. It is able to track the actual load profile and also able to forecast the peak hour loads quite well as compared to the ANN-t. The resulting MAPE is shown in Table 4.2. It can be seen that the overall performance of the ANN-t is vastly improved with the use of the PSO algorithm.

Table 4.3 MAPE performance evaluation for all tested forecasting methods

Day	ANN	ANN-t	ANN-w	ERNN	ERNN-t	ERNN-w	PSO-ANN-t	MLR
Wednesday 9 March 2011	8.05	7.46	6.35	9.22	7.82	7.48	3.52	4.7
Thursday 10 March 2011	6.91	4.40	3.93	9.12	7.56	6.54	2.82	4.1
Friday 11 March 2011	4.32	6.13	5.54	9.25	7.83	7.10	3.66	3.1
Wednesday 6 July 2011	9.52	7.18	6.69	14.01	13.12	14.17	4.47	8.1
Thursday 7 July 2011	4.88	3.05	3.55	11.24	10.83	12.41	2.51	3.4
Friday 8 July 2011	4.07	3.18	4.13	12.04	11.75	12.29	3.71	7.0
Saturday 9 July 2011	8.08	8.27	7.12	6.05	5.92	5.52	4.76	6.9
Sunday 10 July	5.63	9.16	4.98	11.47	10.47	10.82	3.95	9.7
Wednesday 5 October 2011	5.77	6.59	5.08	12.01	11.46	12.27	5.08	6.9
Thursday 6 October 2011	7.55	4.09	10.33	6.00	5.16	6.94	5.70	7.5
Friday 7 October 2011	5.01	4.95	6.65	5.67	4.96	6.63	4.22	5.9
Saturday 15 October 2011	4.75	3.58	4.59	6.23	6.31	6.78	2.77	8.6
Sunday 16 October 2011	10.96	3.47	5.64	10.72	10.66	10.16	3.39	13.3
Average MAPE	6.58	5.5	5.74	9.46	8.76	9.16	3.89	6.86

From Table 4.3, it can be seen that the hybrid network produces the best performance overall with a minimum and maximum forecasting error of 2.51% and 5.70% respectively. Its average error is 3.89% as compared to the rest of the forecasters. One of the objectives of this research was to obtain errors that are ±5%. This was achieved by using a hybrid network for short term load forecasting.

Chapter 5 : Conclusions and Recommendations

Short term load forecasting is essential in electrical utilities as it allows them to control and plan their power system operations. A requirement of this method is that the forecasting tools should forecast as accurately as possible. The degree of error varies from utility to utility and is generally determined by the processes run at that utility. Accurate forecasts can ensure that a utility is able to reduce its generation costs by assisting the operators in making accurate decisions regarding the purchasing of energy as well as scheduling equipment maintenance outages. Large forecasting errors can have an adverse effect on the power system as well as the economic viability of a utility. In this research the acceptable forecasting error was placed at ±5% as the load prediction was for a distribution substation which did not require extremely accurate forecasts as compared to generation forecasting.

The main objective of this research was to develop short term load forecasting models using Computational Intelligence (CI) techniques that incorporated load affecting factors particularly weather. Artificial neural networks (ANN) were utilised to conduct the investigation and the data was split between weekend and weekdays so as to facilitate easy learning. Two forecasters (weekend and weekday) with three variations of inputs as follows were investigated:

- Load, day type
- Load, day type, temperature
- Load, day type, temperature, humidity

The results of the ANN were also then compared to an Elman recurrent neural network (ERNN) in order to determine if better results can be obtained. The same model variations were used for the ERNN. A Mutiple Linear Regression model was also used to forecast using the same data in order to provide a benchmark against traditional forecasting methods. The performance evaluation across all models was conducted using the mean absolute percentage error (MAPE). It was found that the method with the lowest forecasting error and best peak hour (morning and evening) forecasting performance was the temperature sensitive ANN as it was able to forecast with a minimum forecasting error of 3.05% and a maximum forecasting error of 9.16%. The performance of this ANN further proved that temperature plays a major role as a load affecting variable.

A hybrid method consisting of the temperature sensitive ANN and particle swarm optimisation (PSO) was then investigated and tested in an effort to improve the forecasting error to ±5%. The PSO was utilised to alter the weights of the ANN such that the resulting mean square error for the training data was reduced. The performance of the hybrid method was found to be very good as it produced a minimum forecasting error of 2.51% and a maximum of 5.70%. This proves that by introducing hybridization, better forecasting results can be obtained. This work also proves that the problem of low convergence rate of the back propagation method can be overcome by use of PSO algorithm for training purposes.

The observation from this research is that the PSO ANN tended to converge much faster than the ANN and ERNN and also produced good quality results. ERNN converged much faster than ANN however its forecasting accuracy was not acceptable as there were errors in excess of 10%.

5.1 Recommendations

The following further research and studies are recommended in order to be able to further improve on forecasting errors:

- Develop forecasting models for each individual day. It is the author’s belief that it may be possible to further improve forecasts if the neural networks are developed and trained for specific days. A large database of data will be required for this research.
- Include forecasting for holidays (public and possibly school holidays).
- Include more load affecting variables such as cloud cover, wind speed, rainfall, etc.
- Investigate the individual influence of each of the weather variables to determine which is more influential in a load forecast.
- Investigate the use of other techniques such as Support Vector Machine (SVM)to determine the optimal number of hidden neurons.
- Investigate the use of other CI methods in conjunction with ANN such as Population Based Incremental Learning (PBIL). This has not yet been applied to STLF.

References

[1] F.D Galiana, G Gross, “Short term load forecasting”, *Proceedings of the IEEE*, Vol. 75, No 12, 1987, pp. 1558 – 1573

[2] A.D Papalexopoulos, T.C Hesterberg, “A regression based approach to short term load forecasting”, *IEEE Transactions on Power Systems*, Vol. 5, No 4, Nov 1990, pp. 1535 – 1550

[3] W Charytoniuk, M Chen, “Very Short term load forecasting using Artificial Neural Networks”, *IEEE Transactions on Power Systems*, Vol. 15, No 1, Feb 2000, pp. 263 – 268

[4] I Moghram, S Rahman, “Analysis and evaluation of five short term load forecasting techniques”, *IEEE Transactions on Power Systems*, Vol. 4, No 4, Oct 1989, pp. 1484 – 1491

[5] Wikipedia contributors. "Load profile." *Wikipedia, The Free Encyclopedia*, http://en.wikipedia.org/w/index.php?title=Load_profile&oldid=613164305, 16 Jun 2014

[6] F.M Tuaimah, H.M Abdul Abass,”Short-term electrical load forecasting for Iraqi power system based on Multiple Linear Regression method”, *International Journal of Computer Applications (0975 -8887)*, Vol. 100, No 1, Aug 2014, pp. 41 – 45

[7] M.A Farahat, M Talaat, “A New Approach for Short-Term Load Forecasting Using Curve fitting Prediction Optimized by Genetic Algorithms”, *Proceedings of the 14th International Middle East Power Systems Conference (MEPCON’10)*, Cairo University, Egypt, Dec 19-21, 2010

[8] Y Bichpuriya, M.SS Rao, S.A Soman, “Combination Approach for Short Term Load Forecasting”, *9th International Conference on Power and Energy (IPEC 2010) Proceedings*, Singapore, 2010, pp. 818-823

[9] J.P Rothe, A.K Wadhwani, S Wadhwani, “Hybrid and integrated approach to short term load forecasting”, *International Journal of Engineering Science and Technology*, Vol. 2(12), 2010, pp. 7127-7132

[10] A.G Barkitzis, J.B Theocharis, S.J Kiartzis, K.J Satsios, “Short term load forecasting using fuzzy neural networks”, *IEEE Transactions on Power Systems*, Vol. 10, No 3, Aug 1995, pp. 1518–1524

[11] H Shayeghi, H.A Shayanfar, G Azimi, “STLF based on optimized neural network using PSO”, *World Academy of Science, Engineering and Technology*, Vol. 3, 2009, pp. 889 – 899

[12] S Quaiyum, Y.I Khan, S Rahman, P Barman, “Artificial neural network based short term load forecasting of power system", *International Journal of Computer Applications (0975 – 8887)*, Vol. 30, No 4, Sep 2011, pp. 1 – 7

[13] P Subbaraj, V Rajasekaran, “Evolutionary techniques based combined artificial neural networks for peak load forecasting”, *World academy of Science, Engineering and Technology,* Vol. 2, 2008, pp. 600 – 606

[14] A Alshareef,”Next 24 hours load forecasting for the western area of Saudi Arabia using artificial neural network and particle swarm optimization”, *Journal of Engineering and Computer Sciences,* Vol. 3, No 2, Jul 2010, pp. 97 – 117

[15] A Jain, M.B Jain, E Srinivas, “A novel hybrid method for short term load forecasting using fuzzy logic and particle swarm optimisation”, *International conference on power system technology”*, 2010, pp. 1 – 7

[16] P.K Dash, A.C Liew, S Rahman, “Fuzzy Neural-Network and Fuzzy Expert-System for Load Forecasting”, *IEE Proceedings-Generation, Transmission and Distribution*, Vol. 143, No. 1, Jan 1996, pp. 106 – 114

[17] K.H Kim, J.K Park, K.J Hwang, S.H Kim, “Implementation of Hybrid Short Term Load Forecasting System Using Artificial Neural Networks and Fuzzy Expert Systems”, *IEEE Transactions on Power Systems*, Vol. 10, No 3, Aug 1995, pp. 1534-1539

[18] P.K Sarangi, N Singh, D Swain, R.K Chauhan (DR), R Singh (DR), “Short Term Load Forecasting Using Neuro Genetic Hybrid Approach: Results Analysis With Different Network Architectures”, *Journal of Theoretical and Applied Information Technology*, 2009, pp. 109-116

[19] C.S Carlson, W.A Cronje, M.A van Wyk, “Overview of existing methods of load forecasting”, *20th South African Universities Power Engineering Conference*, 13-15 Jul 2011, Cape Town, South Africa, [Discussion Paper]

[20] N Kandil, V Sood, M Saad, “Use of ANNs for Short Term Load Forecasting’, *Proceedings of the 1999 IEEE Canadian Conference on Electrical and Computer Engineering*, 1999, pp. 1057-1061

[21] K.Y Lee, Y.T Cha, J.H Park, “Short term load forecasting using an artificial neural network”, *Transactions on Power Systems*, Vol. 7, No 1, Feb 1992, pp. 124 – 130

[22] E Banda, “Short term load forecasting using Artificial Intelligence techniques”, *Department of Electrical Engineering*, University of Cape Town, Student Thesis, 2006

[23] S Zhang, J Lian, H Xu, J Liu, "Grouping Model Application on Artificial Neural Networks for Short term Load Forecasting", *Proceedings of the 7th World Congress on Intelligent Control and Automation*, 25-27 Jun 2008, pp. 6203-6206

[24] J.K Mandal, A.K Sinha, G Parthasarathy, "Application of Recurrent Neural Network for Short Term Load Forecasting in Electric Power System", *IEEE International Conference on Neural Networks*, Vol. 5, 1995, pp. 2694-2698

[25] N Siddarameshwara, A Yelamali, K Byahatti, "Electricity Short Term Load Forecasting using Elman Recurrent Neural Network", *International Conference on Advances in Recent Technologies in Communication and Computing*, 2010, pp. 351-354

[26] Y Riu, A.A El-Keib, "A Review of ANN-based Short Term Load Forecasting Models", *Proceedings of the 27th South Eastern Symposium on System Theory*, 1995, pp. 78-82

[27] K Liu, S Subbarayan, R.R Shoults, M.T Manry, C Kwan, F.L Lewis, J Naccarino, "Comparison of very short term load forecasting techniques", *IEEE Transactions on Power Systems*, Vol. 11, No 2, May 1996, pp. 877 – 882

[28] W Wu, W Guozhi, Z Yuanmin, W Hongling, "Genetic Algorithm Optimizing Neural Network for Short Term Load Forecasting", *International Forum on Information Technology and Applications*, 2009, pp. 583 – 585

[29] Z.H Osman, M.L Awad, T.K Mahnoud, "Neural Network Based Approach for Short Term Load Forecasting", *Power Systems Conference and Exposition*, 2009, pp. 1- 8

[30] Dr. K Holbert, Fuzzy Logic, http://www.ceaspub.eas.asu.edu/powerzone/FuzzyLogic/index.htm

[31] W Yuill, R Kgokong, S.P Chowdhury, S Chowdhury, "Management of Short Term Load Forecasting in South African Power Networks", *International Conference on Power System Technology*, 2010, pp. 1-8

[32] T.S Dillon, M.A Laughton, "An expert system based load forecasting technique", *Expert system applications in power systems,* Prentice Hall, 1990

[33] A Singh, P.K Kalra, P Emmanual, "A point of view of development of knowledge based systems for load forecasting", *Symposium on expert systems application to power systems*, 1988, pp. 7-1 – 7-4

[34] J.R McDonald, A Asar, W Rattray, “Experience with Artificial Neural Network Models for Short Term Load Forecasting in Electrical Power Systems: A proposed Application of Expert Networks”, *Third International Conference on Artificial Neural Networks*, 1993, pp. 123-127

[35] C.C Liang, K.L Ho, T.S Lai, Y.Y Hau, “Short term Load Forecasting of Taiwan Power System Using A Knowledge-Based Expert System”, *IEEE Transactions on Power Systems*, Vol. 5, No 4, Nov 1990, pp. 1214-1221

[36] D Xin-hui, T Feng, T Shao-qiong, “Study of Power System Short Term Load Forecast Based on Artificial Neural Network and Genetic Algorithm”, *International Conference on Computational Aspects of Social Networks*, 2010, pp 725-728

[37] M Obitko, Genetic Algorithms, http://www.obitko.com/tutorials/genetic-algorithms/crossover-mutation.php , 1998

[38] L Huo, X Fan, Y Xie, J Yin, “Short Term Load Forecasting Based on the Method of Genetic Programming”, *Proceedings of the 2007 IEEE International Conference on Mechatronics and Automation*, Aug 5-8 2007, pp. 839 – 843

[39] Y Yang, G Zheng, D Liu, “BP-GA Mixed Algorithms for Short Term Load Forecasting”, *International Conferences on Info tech and Info net*, Vol. 4, 2001, pp. 334-339

[40] R.C. Eberhart, J. Kennedy, “ A new optimizer using particle swarm theory”, In *Proceeding of the Sixth International Symposium on Micro Machine and Human Science*, Nagoya, Japan, 1995, pp. 39-43

[41] X Hu, Particle Swarm Optimisation, www.swarmintelligence.org/tutorials, 2006

[42] M Hayati, Y Shirvany, “Artificial neural network approach for short term load forecasting for Illam region”, *World Academy of Science, Engineering and Technology*, 2007, pp. 280 – 284

[43] S. Karsoliya, “Approximating number of hidden layer neurons in multiple hidden layer BPNN architecture”, *International Journal of Engineering trends and Technology*, Vol. 3, 2012, pp. 714 – 717

[44] P Sibi, S.A Jones, P Siddarth, ”Analysis of different activation functions using back propagation neural networks”, *Journal of Theoretical and Applied Information Technology*, Vol. 47, No 3, Jan 2013, pp. 1264 – 1268

[45] C Ozkan, F.S Erbek, “The comparison of activation functions for Multispectral Landsat TM Image Classification”, *Photogrammetric Engineering & Remote Sensing*, Vol. 69, No. 11, Nov 2003, pp. 1225–1234

[46] R.C Chakraborty, www.myreaders.info/html/artificial_intelligence.html, 2010

[47] S Tzafestas, E Tzafestas, “Computational intelligence techniques for Short-Term Electric Load Forecasting”, *Journal of Intelligent and Robotic Systems,* Vol. 31, 2001, pp. 7 – 68

[48] L Hernandez *et al.*, “A study of the relationship between weather variables and electrical power demand inside a smart grid/smart world framework”, *Sensors*, 2012, pp. 11571 – 11591

[49] I Nabney, Netlab neural network toolbox version 3.3, www1.aston.ac.uk/eas/research/groups/ncrg/resources/netlab/downloads, 2004

[50] B Sharma, K Venugopalan, “Comparison of Neural Network Training Functions for Hematoma Classification in Brain CT Images”, *Journal of Computer Engineering*, Vol. 16, No 1, Ver. II (Jan. 2014), pp. 31-35

[51] M Orra, G Serpen, “An exploratory study for neural network forecasting of retail sales trends using industry and national economic indicators”, *Proceedings for Workshop on Computational Intelligence in Economics and Finance*, 2005, pp. 875-878

[52] T Gentry, B.M Wiliamowski, L.R Weatherford, “A comparison of traditional forecasting techniques and neural networks”, *Proceedings of Artificial Neural Networks in Engineering*, 1995, pp. 765 – 770

[53] FJ Marin, F Garcia-Lagos, G Joya, F Sandoval, “Global model for short term load forecasting using artificial neural network”, *IEE Proceedings on Generation, Transmission and Distribution,* Vol. 149, No 2, March 2002, pp. 121 – 125

[54] M Hayati, B Karami, “Application of neural networks in short term load forecasting”, *Proceedings of the 7th WSEAS International Conference on Mathematical Methods and Computational Techniques in Electrical Engineering,* 2005, pp. 37 – 41

[55] J. Kennedy, R. C Eberhart, “Particle Swarm Optimization”, *Proceedings of IEEE International Conference on Neural Networks*, volume IV, Perth, Australia, 1995, pp. 1942-1948

APPENDIX A

1. List of publications

Aspects of this work have been presented at the following peer reviewed conferences:

1. E Banda, K.A Folly, " A Review on Artificial Intelligence Techniques used for Short Term Load Forecasting", *Proceedings of Southern African Universities Power Engineering Conference (SAUPEC 2011),* Cape Town (RSA), Jul, 2011
2. E Banda, K.A Folly, "Day ahead load forecasting using an Atificial Neural Network and Elman Recurrent Network", *Proceedings of International Association of Science and Technology for Development (IASTED 2012),* Gaborone (Botswana), Aug, 2012
3. E Shezi, K.A Foll, "Short term load forecasting using PSO and ANN", *Proceedings of Southern African Universities Power Engineering Conference (SAUPEC 2015),* Johannesburg (RSA), Jan, 2015