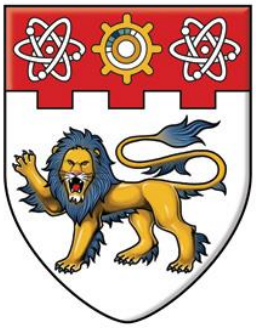


NANYANG TECHNOLOGICAL UNIVERSITY

ENERGY FORECASTING FOR POWER SYSTEMS

LI SONG

School of Electrical & Electronic Engineering

Nanyang Technological University

2015

Energy Forecasting for Power Systems

Li Song

School of Electrical & Electronic Engineering

A thesis submitted to Nanyang Technological University

in fulfillment of the requirement for the degree of

Doctor of Philosophy

2015

Acknowledgments

I would like to express my sincere gratitude to my supervisor, Prof. Lalit Goel for his guidance, advices and support throughout my PhD candidature. His attitude and enthusiasm toward work and life not only help me during my PhD study but also will always benefit me.

I would also like to express many thanks and appreciations to Assoc Prof. Wang Peng for his knowledge, encouragement and instructions. His suggestions on research method and writing skills of academic articles assist me to complete my degree successfully. His advices for career are highly appreciated.

My gratitude is then expressed to the research students and technical staffs in Lab of Clean Energy Research, NTU. We have spent a lot of time together. A special note of thanks goes to Mr. Foo Mong Keow, Thomas and Ms. Chia-Nge Tak Heng for their technical assistance.

Finally, taking this opportunity, I would like to express my gratitude to my family for their endless love and support, from the bottom of my heart.

Table of Contents

Summary

Energy forecasting has always been an essential part in the operation and planning of power systems. Accurate and reliable forecast models allow electric utilities to make timely decisions and carry out effective manipulations. Energy forecasting is becoming more important due to the revolutionary changes taking place in power systems, such as the promotion of smart grid technologies, the deregulation of electricity markets and the penetration of renewable energies. Although a large number of methods have been tried out on energy forecasting, there is still a lot of room for improvement. In this thesis, the research focus is on developing advanced models for electric load and wind power forecasting.

Short-term load forecasting (STLF) refers to the estimation of electric load demand from one hour ahead up to a week ahead. Load forecasting plays a very important role in many fields of power systems, such as energy market analysis, power generation scheduling, unit commitment and security assessment. Precise forecasting results can help to improve the power system reliability, reduce the operating cost and cut down the occurrences of power interruption events. In this thesis, the background of STLF has been presented and the state-of-the-art approaches for STLF have been reviewed.

In view of the key findings from literature review, this thesis has proposed two novel STLF methods based on artificial neural networks (ANNs). The first method combines a special ANN called extreme learning machine, wavelet transform and a modified artificial bee colony algorithm. The second method is an ensemble forecaster, in which wavelet transform, hybrid ANNs, input feature

selection and partial least squares regression are involved.

Wind power, one of the most popular renewable energies, has established itself as a promising supplement for electric power generation. The penetration of wind power would produce not only economical and environmental benefits but also uncertainty and intermittency to power systems. Precise wind power forecasting (WPF) approaches are therefore imperative for power industries. In this thesis, an ensemble of ANNs with input feature selection is proposed to forecast the power generation of a wind farm.

To confirm their effectiveness and superiority, the proposed methods have been tested using actual electrical load and wind power data. Numerical results reveal that the proposed methods can obtain better forecasting performance than other standard and state-of-the-art methods. Some directions for future research have also been identified.

List of Figures

List of Tables

List of Abbreviations

PV	Photovoltaic
VSTLF	Very Short-Term Load Forecasting
STLF	Short-Term Load Forecasting
MTLF	Medium-Term Load Forecasting
LTLF	Long-Term Load Forecasting
ELM	Extreme Learning Machine
THI	Temperature Humidity Index
WCI	Wind Chill Index
MLR	Multiple Linear Regression
AR	Autoregressive
MA	Moving Average
ARMA	Autoregressive Moving Average
ARIMA	Autoregressive Integrated Moving Average
ARMAX	Autoregressive Moving Average with Exogenous Variables
ARIMAX	Autoregressive Integrated Moving Average with Exogenous Variables
AI	Artificial Intelligence
ANN	Artificial Neural Network
SVM	Support Vector Machine
MSE	Mean Square Error
MAPE	Mean Absolute Percentage Error
MAE	Mean Absolute Error
RMSE	Root Mean Square Error
NRMSE	Normalized Root Mean Square Error

WPF	Wind Power Forecasting
NWP	Numerical Weather Prediction
ITSM	Improved Time Series Method
RNN	Ridgelet Neural Network
GP	Gaussian Process
BP	Backpropagation
SLFN	Single-Hidden Layer Feedforward Neural Network
ABC	Artificial Bee Colony
MABC	Modified Artificial Bee Colony
MCN	Maximum Cycle Number
MP	Moore-Penrose
WNN	Wavelet Neural Network
SIWNN	Similar Day-Based Wavelet Neural Network
ESN	Echo State Network
PLSR	Partial Least Squares Regression
CMIFS	Conditional Mutual Information Based Feature Selection
LM	Levenberg-Marquardt
PRESS	Prediction Residual Sum of Squares
MI	Mutual Information
CMI	Conditional Mutual Information
PM	Parallel Model
CA	Correlation Analysis
RBFNN	Radial Basis Function Neural Network
LMNN	Neural Network with Levenberg-Marquardt
mRMR	Minimum Redundancy Maximum Relevance

Chapter 1 Introduction

1.1 Electric Power System

Electricity is one of the great discoveries that have changed the daily lives of human beings. The demand for electricity has experienced a tremendous rise due to the continuous expansion of world population and the rapid development of global economy. The international energy outlook 2013 (IEO2013) projected that the world net electricity generation will increase by 93%, from 20.2 trillion kWh in 2010 to 39.0 trillion kWh in 2040 (Figure 1.1) [1].

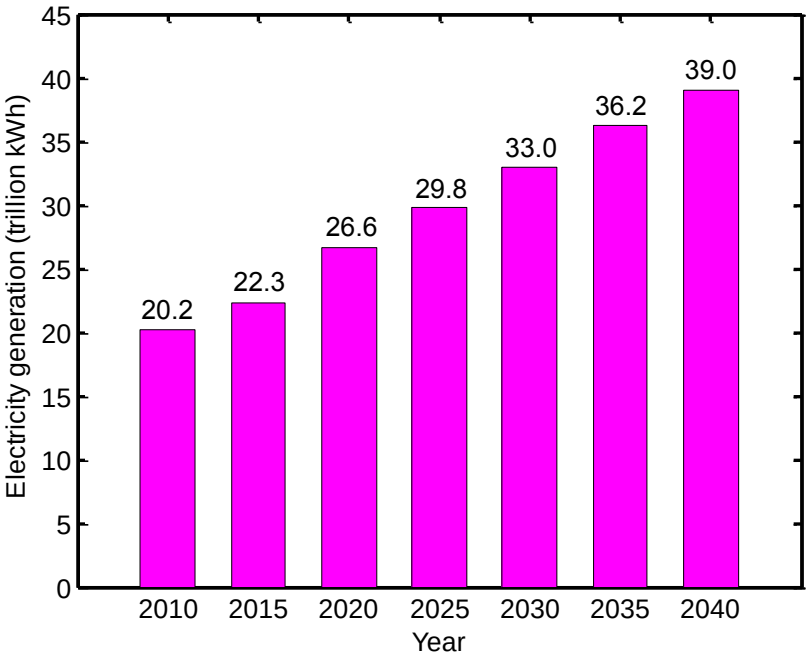


Figure 1.1 World net electricity generation projections [1].

Electricity cannot be used on a large scale without electric power systems. As shown in Figure 1.2, an electric power system comprises three distinct parts: generation, transmission and distribution. In the generating station, most of the electricity is produced by three-phase synchronous machines, which convert the primary energy contained in raw fuels (e.g. coal and natural gas) to mechanical energy and finally to electrical energy. The generated electricity is transmitted over significant distances to consumers through a complex network of electrical components, including transformers and overhead transmission lines. The step-up transformers are deployed to increase the voltage from generating levels to transmission levels and the step-down transformers are utilized to decrease the voltage from transmission levels to distribution levels. The distribution network is the final phase in the transmission of electricity to individual consumers. The consumers may be classified as industrial, commercial and residential. They are supplied via feeders at various voltage levels with respect to their requirements.

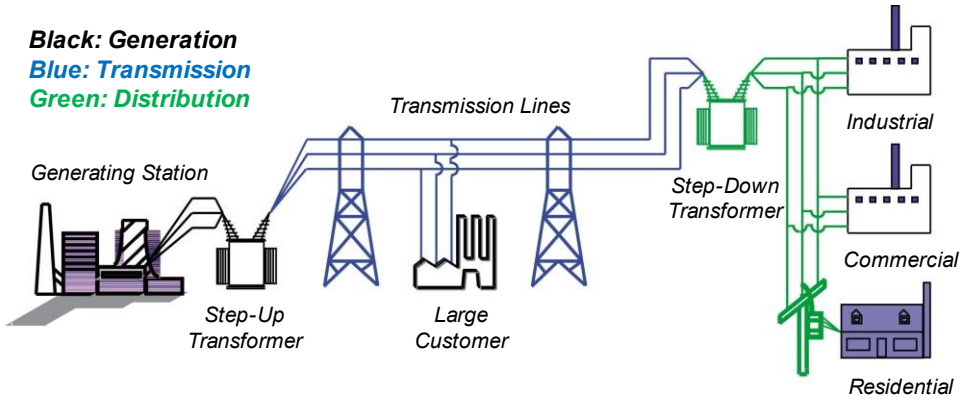


Figure 1.2 Structure of electric power system.

In recent times, revolutionary changes have taken place in electric power systems, which present numerous opportunities and challenges:

a) Smart Grid

Smart grid, well-known as the next generation electricity grid, uses information and communication technology to improve the utilization of electrical energy in four aspects: efficiency, reliability, sustainability and economy [2]. The major differences between the existing grid and smart grid are given in Figure 1.3. In order to realize the smart grid, the technologies in information and communications, monitoring systems, power inverters and converters, computer hardware and software, smart meters, energy storage, data management systems, etc., have attracted wide attention.

Existing grid	Smart grid
• Electromechanical	• Digital
• One-way communication	• Two-way communication
• Centralized generation	• Distributed generation
• Hierarchical	• Network
• Few sensors	• Sensors throughout
• Blind	• Self-monitoring
• Manual restoration	• Self-healing
• Failures and blackouts	• Adaptive and islanding
• Manual check/test	• Remote check/test
• Limited control	• Pervasive control
• Few customer choices	• Many customer choices

Figure 1.3 Major differences between existing grid and smart grid [3].

b) Electricity Market Deregulation

Electricity can be sold, bought and traded in the market. Traditionally, the electric utilities managed the entire electric system: generation, transmission and distribution in their service zones. A

customer who needs electricity has to buy it from the nearest utility. In the deregulated electricity market, the functions of a traditional utility are separately realized by different participants: generation companies, transmission companies, distribution companies and retail companies [4]. Under deregulation, the competition in the electricity industry is expected to increase, which is beneficial to reduce electricity price, attract investment and provide customers more options. A customer can play a much more proactive role in the deregulated market, like selling excess electric power produced by distributed generators to others or providing demand response capacity to utilities [5].

c) Renewable Energy

Renewable resources such as wind, sunlight, tides and geothermal heat are promising alternatives to fossil fuels (e.g. coal and natural gas) for generating electric power. Firstly, it is not necessary to worry about the shortage issue of renewable resources because they are continually replenished. It is recognized that all the renewable resources, regardless of their forms, derive energy either directly or indirectly from the sun. Secondly, compared to fossil fuels, some of the renewable resources are ubiquitous on the planet. For example, sunlight is available everywhere on earth and more importantly it is totally free for us to harvest and use. Thirdly, renewable resources are eco-friendly and would not pollute the environment. For instance, coal-fired power plants usually need a large amount of water for cooling and discharge various pollutants (nitrogen oxides and heavy metals) into the eco system. In contrast, photovoltaic

(PV) systems do not have such problems while converting solar energy to electricity.

The contribution of renewable energy has progressed substantially. In 2013, renewable energy supplied about 27% of global power generation [6]. By the end of 2013, the capacity of renewable energy (including hydro power) has surpassed 1560 GW, yielding a staggering increase (95%) over 2004 [7]. The installed capacities of solar PV, wind, biomass and geothermal power between 2004 and 2013 are illustrated in Figure 1.4. It can be observed that all the renewable energy technologies have shown improvements during this decade. More specifically, solar PV achieved the largest increase rate (5246%) among all technologies from 2004 to 2013, while wind power made the biggest increment (270 GW) of installed capacity during the same time.

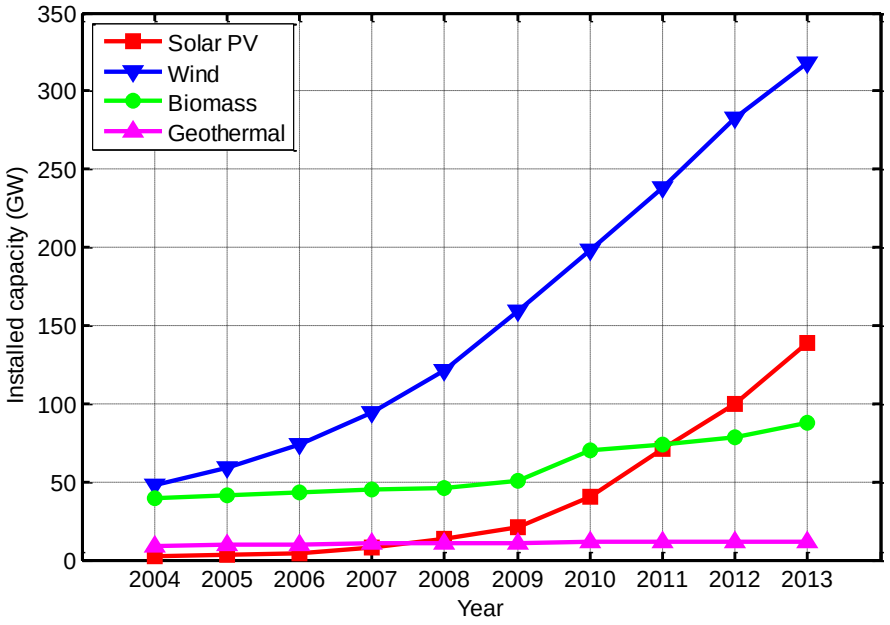


Figure 1.4 Installed capacities of solar PV, wind, biomass and geothermal power.

Renewable energy markets have expanded rapidly in recent years. A large number of countries are involved in the promotion of renewable technologies. By 2013, over 140 countries announced their policies and targets to support the research and development of renewable technologies, which offer a strong driving force to spread the renewable markets. Renewable technologies can not only continuously contribute in energy generation but also lead to economic benefits by reducing the quantity of imported fuels, improving air quality, creating job positions, etc. Many scientists are committed to working on the synergy between energy efficiency and renewable energy technologies. It can be asserted that renewable energy will play an increasingly important role towards a sustainable earth.

However, the addition of renewable energy would also cause some difficulties in power system operation and planning, mainly because of the uncertain and intermittent nature of renewable sources. Renewable energy is regarded as non-dispatchable and the significant variation of renewable energy generation makes it difficult to maintain the balance between power supply and load demand. As a result, the power system operators must rethink their generation plans and reserve capacities.

1.2 Motivations

Energy forecasting has always been an essential element in power systems operation, control and planning. In most electric utilities, energy forecasting is integrated with the energy management system. Accurate and reliable forecast results help the utilities to make timely decisions and carry out effective

manipulations. Energy forecasting is becoming more important due to the tremendous changes taking place in the existing power systems, such as the promotion of smart grid, the deregulation of electricity markets and the penetration of renewable energy. In general, energy forecasting is imperative in, but not limited to, the following fields:

1) Operation and planning: Energy forecasting results can be used to make vital decisions, including power generation control, economic dispatch, load switching, and voltage regulation. Suitable decisions like these can balance electricity supply and demand, improve power system security, reduce operating costs and decrease occurrences of failure events. For example, if the upsurge of wind power output is known in advance, we can schedule the shutdown of more expensive natural gas-fired plants.
2) Energy purchasing and bidding: Energy price fluctuates frequently in a competitive energy market. Energy forecasting helps to evaluate energy purchasing agreements, make bids and decrease the opportunity costs.
3) Maintenance and construction: Energy forecasting can be harnessed to save costs when a utility schedules maintenance outages and constructs extra infrastructure. It is obvious that the coal-fired power plants, wind turbines, transformers and other devices require regular inspections and maintenance. Moreover, due to the growing population and economic development, the utilities have to reinforce network, build substations, etc., to meet the rising energy demand, all of which rely heavily on energy forecasting.
4) Demand response: Smart grid technologies drive the energy consumers

to be more active in the realm of energy management. Instead of adding more generators to meet demand, demand response offers an alternative that the consumers can voluntarily reduce their electricity consumption, like turning off heating or cooling system. The participants will get paid from utilities for providing demand response capacity. The utilities will also benefit from the demand response programs because this solution is more efficient and economical. It is evident that the load forecasts for buildings, houses and even devices are necessary in such programs.

1.3 Objectives

The real challenge in energy forecasting is to make forecasts as accurate as possible. So far, many techniques have been tried out and plenty of methods have been proposed for energy forecasting, with varying degrees of success. But the evolving power systems never cease to demand more accurate forecasting models. In this thesis, the research focus is on developing advanced models for short-term load and wind power forecasting.

Short-term load forecasting (STLF) refers to the estimation of electric load consumption by end users from one hour ahead up to one week ahead. STLF is a traditional but very important topic in power systems, which has been studied in thousands of articles. The development of an accurate STLF model requires an in-depth understanding of load characteristics to be modeled. This knowledge is typically formed from heuristic experience as well as statistical analysis. In this thesis, two new STLF methods have been developed based on artificial neural networks. In the first method, extreme learning machine as an

emerging class of neural networks is employed as the forecast engine. Two supporting techniques, wavelet transform and a modified artificial bee colony algorithm, are adopted to assist the extreme learning machine. The second method refers to an ensemble forecaster, which integrates a novel wavelet-based ensemble strategy, extreme learning machine and partial least squares regression. Moreover, an innovative hybrid learning algorithm and a robust feature selection criterion are developed to further improve the forecasting performance.

Wind power, one of the most prevalent renewable energy technologies, has established itself as a promising supplement for electric power generation. The increasing penetration of wind power would create not only environmental and economic benefits but also uncertainties (induced by the intermittency of wind) to the existing power grid. The only way to integrate wind power to the existing power grid is to establish a wind power forecasting system. This thesis proposes an ensemble of artificial neural networks with input feature selection to forecast the power generation of a wind farm.

1.4 Major Contributions

This thesis proposes several advanced models based on machine learning methods to improve short-term load and wind power forecasting performance. These models may be used to address numerous challenges in power systems. The major contributions of this thesis are summarized as follows:

1) A hybrid STLF model is proposed based on extreme learning machine (ELM). Two improvements are conducted to tackle two major problems

in STLF: the nonstationary behavior of load data and the robustness of forecasting model. Firstly, the wavelet transform is used to decompose the load series into several sub-components with different frequencies. Each sub-component of the load series is then separately forecasted by an ELM-based predictor. Secondly, a modified artificial bee colony (MABC) algorithm with strong global search capability is developed to search for the optimal parameters of input weights and hidden biases for ELM. Compared to the conventional neuro-evolution method, the proposed ELM-MABC method can yield better learning accuracy with fewer iterative steps.

2) An innovative ensemble method is proposed for STLF, which integrates wavelet transform, neural networks, input variable selection and partial least squares regression. Firstly, a novel wavelet-based ensemble strategy is presented to generate a collection of individual forecasters. The ensemble strategy can take advantage of the useful complementary information to improve the generalization capability. Secondly, a new feature selection criterion based on conditional mutual information is developed to choose a compact set of informative input variables. The irrelevant, weakly relevant and redundant input variables are removed at this stage. Thirdly, a hybrid learning algorithm merging Levenberg-Marquardt and extreme learning machine is developed to improve the learning performance of neural networks. The Levenberg-Marquardt algorithm is a very efficient gradient-based optimization method, which can approach the second-order training speed without calculating the

Hessian matrix [8]. Finally, partial least squares regression is applied to combine the individual outputs to make an accurate ensemble forecast, which can identify the unique contribution of each individual forecast.

3) A novel ensemble method consisting of neural networks, wavelet-based ensemble strategy, feature selection and partial least squares regression is proposed for the generation forecasting of a wind farm. The neural networks are trained using the Levenberg-Marquardt leaning algorithm. The ensemble strategy and feature selection method are borrowed from the previous STLF approach and fine-tuned to meet the requirements of wind power forecasting. Partial least squares regression is again used to establish an accurate ensemble forecast.
4) The proposed forecasting methods are tested using actual load and wind power data. Numerical results confirm that the proposed methods are able to surpass other standard and state-of-the-art models. This implies that the proposed energy forecasting methods show a great potential for practical applications.

1.5 Organization

The reminder of this thesis is organized as follows.

Chapter 2 introduces the fundamentals of load and wind power forecasting, including classifications and influential factors. The state-of-the-art methods for short-term load and wind power forecasting are reviewed.

Chapter 3 presents a novel method for short-term load forecasting based on wavelet transform and evolutionary extreme learning machine. Particulars

about wavelet transform, extreme learning machine and artificial bee colony are described. The merits of evolutionary extreme learning machine are depicted.

Chapter 4 presents a novel ensemble model for short-term load forecasting. The existing ensemble approaches for short-term load forecasting are discussed. The advantages of the proposed wavelet-based ensemble strategy are presented. A new hybrid learning algorithm combining Levenberg-Marquardt and extreme learning machine is developed. The step-by-step procedure of proposed feature selection criterion is summarized. A comparison of the proposed input feature selection method and other benchmark methods is conducted.

Chapter 5 presents a novel ensemble approach for wind power forecasting, which involves neural networks, feature selection, wavelet transform and partial least squares regression. The proposed method is tested based on actual data from two wind farms.

Chapter 6 concludes this thesis and recommends some research directions.

Chapter 2 Literature Review

2.1 Introduction

Accurate models for load and wind power forecasting are vitally important for electric utilities [9-11]. In the past few decades, a number of forecasting methods have been developed, with varying degrees of success. The forecasting process can range from very simple to very complex. In this chapter, a comprehensive literature review for short-term load and wind power forecasting is presented.

2.2 Electric Load Forecasting

2.2.1 Overview

Electric load forecasting refers to the estimation of future load demand. A typical forecasting process involves modeling the relationship between load and its influential factors, such as temperature, wind speed and day type [12]. As shown in Figure 2.1, load forecasting can be accomplished by three steps:

1) Identify the influential factors that affect the load consumption.
2) Quantify the underlying relationship between load and its factors using a suitable mathematical method.
3) Extrapolate the load forecast by including the most recent information.

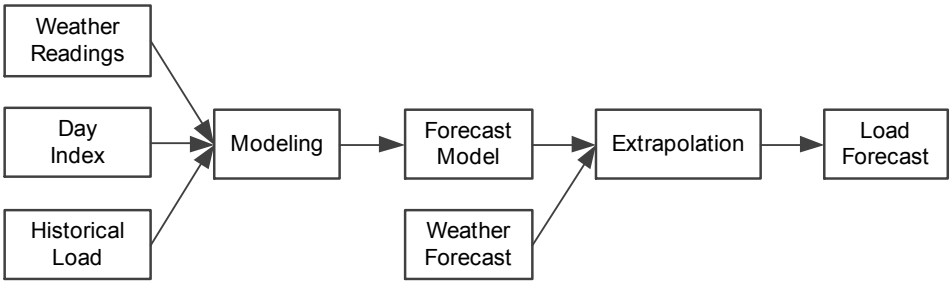


Figure 2.1 Load forecasting process.

Load forecasting can be roughly classified into four categories based on its forecasting horizon: very short-term load forecasting (VSTLF), short-term load forecasting (STLF), medium-term load forecasting (MTLF) and long-term load forecasting (LTLF) [13]. It should be pointed out that the horizon ranges are not strictly defined. Different load forecasts have different applications in power systems, as summarized in Table 2.1.

Table 2.1 Load forecasting categories

Category	Forecasting horizon	Applications
VSTLF	Few minutes to one hour	Optimal power flow State estimation Real-time contingency analysis
STLF	One hour to one week	Unit commitment Economic dispatch
MTLF	One week to one year	Reserve requirement decision Maintenance scheduling
LTLF	One year and above	Infrastructure development Financial planning

2.2.2 Influential Factors for Load Forecasting

In general, there are four types of factors that affect the load demand [14]:

a) Weather Conditions

Weather conditions are usually the most significant factors to drive the fluctuations of load consumption. The effects of weather conditions can be quantified by finding a linear or nonlinear relationship between load and weather variables. A number of weather variables are available for use, but it is not necessary to take all of them as inputs. Much effort has been spent on selecting the most important ones, which contribute to the majority share of weather-sensitive load. These factors consist of temperature, humidity and wind speed.

In most situations, temperature is the most dominant factor, which accounts for the largest piece of weather-sensitive load. The variations of temperature result in the changes of load demand [15]. For example, the temperature drop in winter would lead to an increase in the heating load, and the temperature rise in summer would boost the cooling load. The relationship between load and temperature is shown in Figure 2.2. The hourly data are taken from ISO New England, an independent system operator in USA [16]. The load values are the measured system loads of ISO New England. The time period is from January 1, 2010 to June 31, 2010, including winter, spring and summer. It is obvious that there exists a nonlinear relationship between load and temperature. A piecewise linear function is also given to model the load-temperature relationship. This function has a minimum point where the temperature is around 57 °F. Many papers have studied the load-temperature relationship. For example, a piecewise linear function was described in

[17]. In [18], a third order polynomial fitting function was proposed.

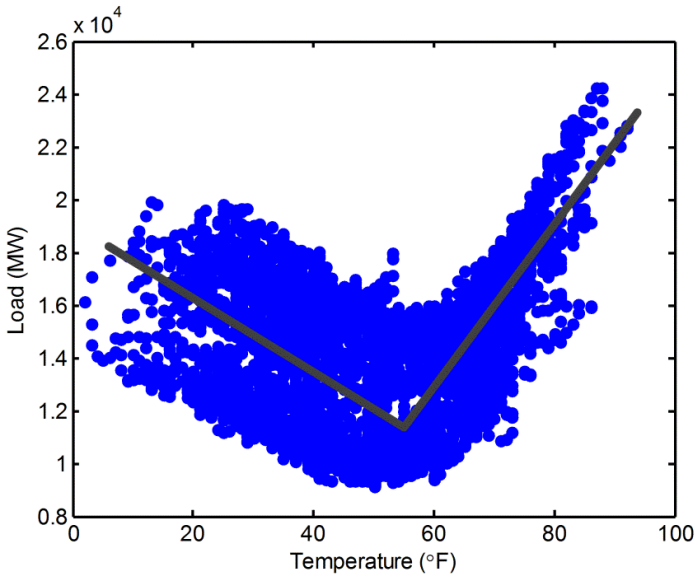


Figure 2.2 Load versus temperature.

It is interesting to note that the effect brought about by temperature can also be modeled by fitting a function between load and temperature deviation. It is based on the fact that the load demand only varies when the temperature deviates from the normal value.

Humidity is another key factor that affects the variations of cooling load, especially in warm weather. People tend to feel uncomfortable in high humidity environment. The effect on cooling load can be stated as a function of a humidity measure, such as temperature humidity index (THI), relative humidity and dew point temperature.

Wind speed is also an important factor that influences the weather-sensitive load. The cooling effect from wind is subject to wind speed as well as temperature. Wind chill index (WCI) is commonly employed to

represent the wind cooling effect in load forecasting models. Figure 2.3 presents a piece of wind speed data, which are collected by the National Renewable Energy Laboratory [19]. It can be seen that the wind speed series is very chaotic.

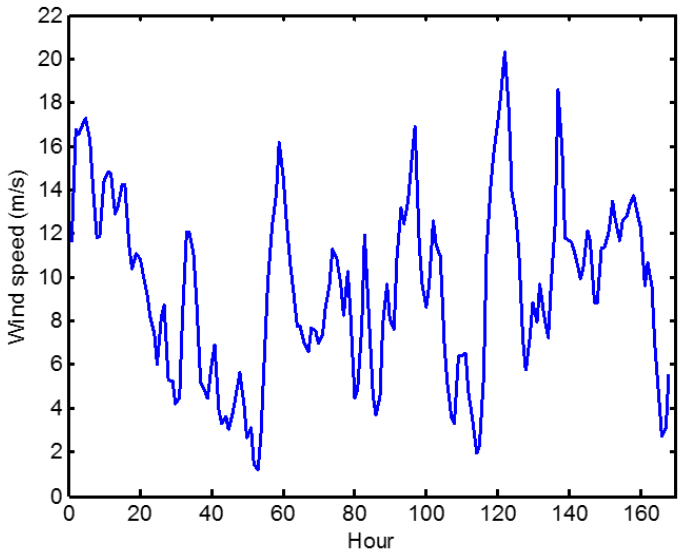
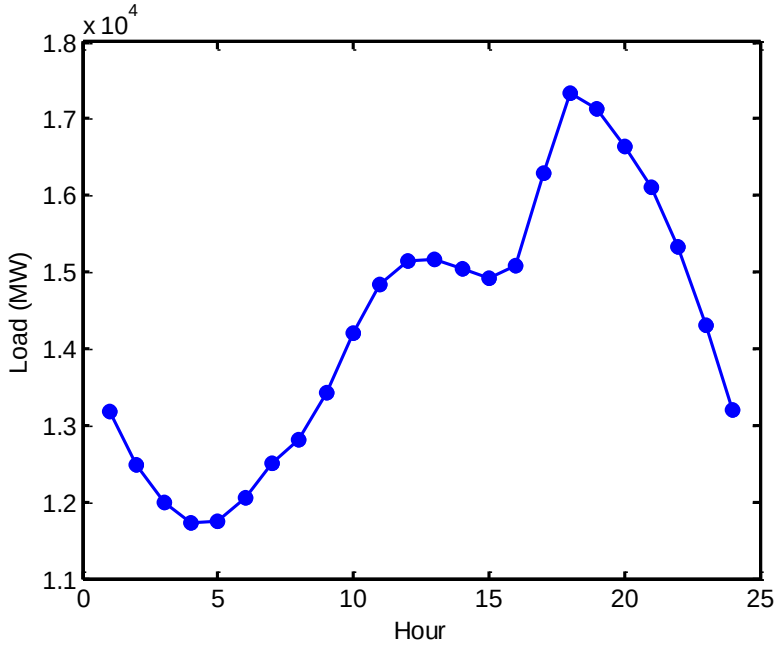


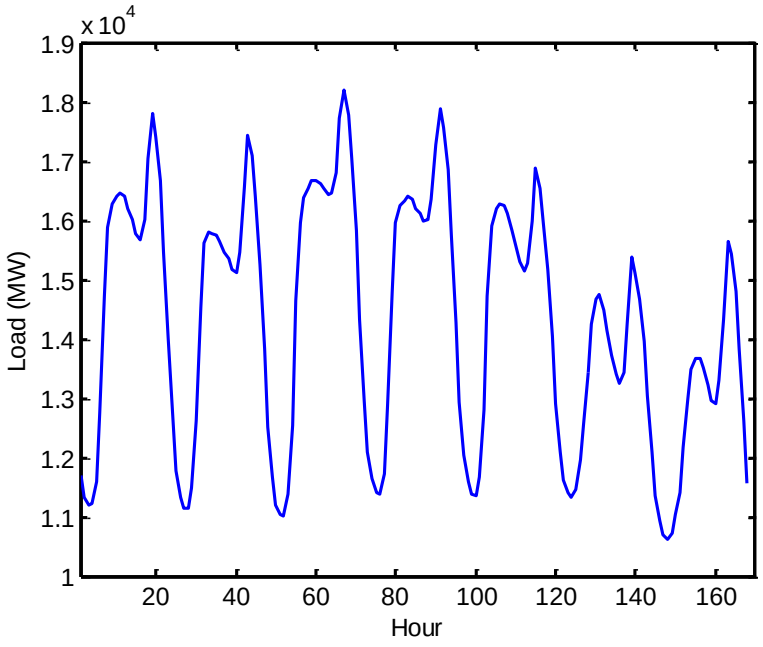
Figure 2.3 Wind speed data in a week.

b) Calendar Variables

The calendar variables determine the shape of load series. The load demand can vary significantly from day to night. It also exhibits weekly as well as seasonal periodicities. This is because human activities have substantial influences on load trends. The calendar variables consist of the month of the year, the day of the week and the hour of the day. The actual load measurements from ISO New England are shown in Figure 2.4. The selected day is January 1, 2010 (Figure 2.4a) and the chosen week is from March 1, 2010 to March 7, 2010 (Figure 2.4b).



(a) Load behavior in a day.



(b) Load behavior in a week.

Figure 2.4 Two segments of load data.

It can be seen that the load behavior in the daytime differs greatly

from that in the night. The daily peak load occurs around 6 p.m. For the weekly behavior, the load on weekdays is bigger than that of weekends. In addition, the load of public holidays is relatively difficult to forecast because of their irregular occurrences.

c) Types of Customers

As shown in Figure 1.2, the customers on the demand side can be classified into three groups: residential, commercial and industrial. The usage pattern of electricity is different for each type of customers [10]. Industrial customers normally exhibit uniform load demand compared with the other two types of customers. The air-conditioning and lighting loads of residential and commercial buildings are the primary driving forces of peak demand of electricity. The different customers need to be modeled separately using specific variables with regard to their usage patterns [10].

d) Special Events

Some special events like strikes, earthquakes, political conventions and popular TV programs can trigger large deviations on the demand of electricity. These factors should also be considered in load forecasting.

2.2.3 Load Models

In order to build the forecast model, one can express the load demand as a function of a set of measurable variables associated with the influencing factors. The parameters of the forecast model are usually determined via statistical analysis of the historical data. These models, together with the most recent

information (e.g. weather forecasts), are deployed to forecast future load values. The two categories of such models are: additive model and multiplicative model.

a) Additive Model

In the additive model, the total load demand for the system is made up of four components [13]:

$$L(t) = L_b(t) + L_w(t) + L_s(t) + L_r(t) \tag{2.1}$$

where $L(t)$ is the system load at time t, $L_b(t)$ is the base or normal load which is based on the economic circumstances of the area, $L_w(t)$ is the weather-sensitive load component, $L_s(t)$ is the load increment due to the special events and $L_r(t)$ is the load term associated with random factors in power systems.

b) Multiplicative Model

A multiplicative load model may be given by

$$L(t) = L_b(t) \cdot F_w(t) \cdot F_s(t) \cdot F_r(t) \tag{2.2}$$

where $L(t)$ is the system load and $L_b(t)$ is the base load. $F_w(t)$, $F_s(t)$ and $F_r(t)$, known as correction factors, are three positive numbers to scale the base load. Specifically, $F_w(t)$ represents weather correction, $F_s(t)$ represents special events correction and $F_r(t)$ represents random factors correction.

2.2.4 Review of Short-Term Load Forecasting Methods

Over the past forty years, numerous technical papers have been published

to address the load forecasting problem. In practice, the methods for medium- and long-term forecasting are different from those for short-term forecasting. In this thesis, our attention is focused on the short-term load forecasting problem, unless otherwise stated.

Short-term load forecasting (STLF) is concerned with the prediction of the system load demand over a period ranging from one hour to one week [9]. The primary application of STLF is to provide load predictions for power generation scheduling. A secondary application of STLF is for the predictive assessment of power system security. Generally speaking, the STLF methods presented in the literature can be classified into two groups: statistical methods and artificial intelligence methods. Statistical methods include linear regression, time series, exponential smoothing, state-space model and Kalman filtering, whilst artificial intelligence methods consist of artificial neural networks, expert systems, fuzzy logic and fuzzy systems, support vector machines, etc.

a) Similar Day Approach

Similar day approach is a naïve attempt for STLF, which involves searching the previous data for days with similar features (e.g. weather conditions or calendar indicators) to the forecast day. It is based on the assumption that the load of a similar day is considered as a reasonable forecast. One can use a single similar day. Alternatively, to improve the forecasting accuracy, several similar days can be selected and sent to a forecast engine (e.g. linear regression). In [20], Euclidean norm with weighted factors was used to quantify the similarity between a forecast day and a searched historical day. In this case, smaller Euclidean norm

meant better similarity.

b) Linear Regression

Multiple linear regression (MLR) is a simple but effective method for STLF. In the MLR method, the load is expressed as a function of its explanatory variables such as weather and non-weather variables which affect the electrical load. The load model using MLR is given by [21]

$$y(t) = a_0 + a_1 x_1(t) + a_2 x_2(t) + \cdots + a_p x_p(t) + e(t) \tag{2.3}$$

where $y(t)$ is the system load, $x_1(t),\ldots x_p(t)$ are the explanatory variables, a_0, $a_1,\ldots$, a_p are the regression coefficients and $e(t)$ is the residual load. Typically, the regression coefficients are estimated using a least squares method. As shown in (2.3), for each unit increase in the value of x_p, the target variable y will increase by a_p units. However, in many situations, such a linear relationship may not hold. Then, we can use a polynomial regression model.

In a polynomial regression model, the load y is expressed as an nth degree polynomial of the explanatory variable x by [18]

$$y(t) = a_0 + a_1 x(t) + a_2 x^2(t) + \cdots + a_n x^n(t) + e(t). \tag{2.4}$$

It is pointed out that (2.4) is a linear model since the regression function is linear with respect to the regression coefficients a_0, $a_1,\ldots$, a_n. In fact, (2.4) is a special case of MLR model if the terms x, $x^2,\ldots$, x^n are treated as distinct independent variables.

Linear regression has been widely utilized for STLF, because of its simple implementation. A naïve MLR benchmark was proposed in [22],

in which the interaction effects between the explanatory variables were considered. This MLR model has served as a benchmark in a US utility for in-hour STLF. In [23], several MLR models were used to predict the peak load of the next day. In [24], a regression based peak load model with a transformation technique was described. The transform function was founded to deal with the daily and seasonal variations in load data. In [18], a third-order polynomial was proposed to fit the relationship between load and temperature.

c) Time Series Approach

Time series approach is developed to consider the possible internal structure in the load series and extract some statistical features, such as autocorrelation, trend and seasonal variations. The autoregressive (AR) and moving average (MA) are the two basic time series models [25].

The AR model of order p, denoted by AR(p), is given by

$$y(t) = a_1 y(t-1) + a_2 y(t-2) + \cdots + a_p y(t-p) + e(t) \qquad (2.5)$$

where $y(t)$ is the system load at time t, a_0, a_1,…, a_p are the coefficients and $e(t)$ is the random disturbance. It is seen that the current load value is expressed as a linear combination of p past load values and a random disturbance. The MA model of order q, denoted by MA(q), is given by

$$y(t) = e(t) + b_1 e(t-1) + b_2 e(t-2) + \cdots + b_q e(t-q) \qquad (2.6)$$

where the current load value is expressed as a linear combination of the current and previous random disturbances. The autoregressive moving average (ARMA) model is a mixture of AR and MA, which is given by

$$y(t) = \sum_{i=1}^{p} a_i y(t-i) + \sum_{j=1}^{q} b_j e(t-j) + e(t) \tag{2.7}$$

The above three models are all stationary time series processes, i.e. the mean and variance do not shift over time and do not follow any trends. As load series is nonstationary, neither of them can yield very accurate forecasts alone. However, they form the basis to develop advanced time series models.

A nonstationary process may be transformed to a stationary one by differencing. This leads to an autoregressive integrated moving average (ARIMA) model, which is a generalization of ARMA. The absence of weather variables in time series models would damage the forecasting accuracy, especially when there are sudden changes in weather. Hence, autoregressive moving average with exogenous variables (ARMAX) and autoregressive integrated moving average with exogenous variables (ARIMAX) models are developed [25].

The early application of time series approaches to load forecasting was reviewed in [18]. This paper noted that Box and Jenkins time series models were suitable to the load forecasting problem. Moreover, a third order polynomial was used to represent the nonlinear load-temperature relationship and integrated into the ARIMA model. A novel time series model was proposed in [26], which can model the valuable experiences of the expert operators. It was shown that this method can provide more accurate forecasting results than artificial neural networks and Box and Jenkins models. In [27], a new self-organizing fuzzy ARMAX approach

was proposed for day ahead hourly load forecasting. Some other time series based methods for load forecasting were reported in [28, 29].

d) Artificial Neural Networks

The models discussed so far are reliable and attractive, since some physical interpretation can be attached to their components. However, they are basically linear approaches and the daily load is known to be a nonlinear function of the exogenous variables [9]. The linear methods are incapable of adapting to sudden weather changes and irregular holiday activities [30]. In the presence of such events, the load forecasts are not as satisfactory as desired.

In recent times, artificial intelligence (AI) techniques such as artificial neural networks, expert systems, and support vector machines have been widely used to solve the load forecasting issue. AI methods have shown a strong ability to output good performance in modeling nonlinear load series. The main advantage of AI methods is that they do not require any postulate models between inputs and outputs [30].

An ANN contains a massive number of interconnected information processing units, called artificial neurons. The interconnections, known as synaptic weights, are applied to learn the knowledge from the environment and make it available for use. The neurons are usually organized into three kinds of layers: input layer, hidden layers and output layer. A feedforward neural network is illustrated in Figure 2.5. This fully connected network is made up of an input layer with multiple nodes, a hidden layer with a desired number of nodes and an output

layer with a single node.

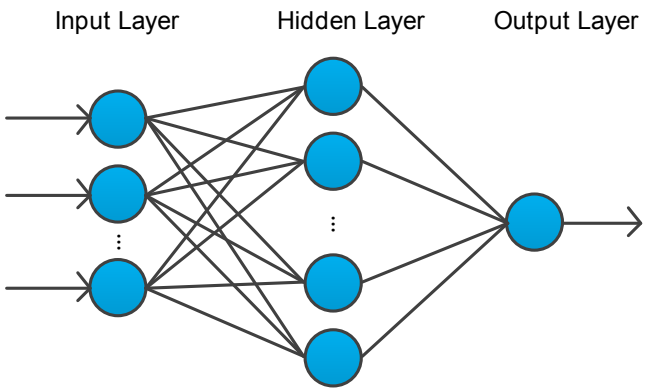


Figure 2.5 Feedforward neural network.

ANN is very suitable to model the nonlinear relationship between the load profile and its dependent factors. Theoretically, it has been shown that ANN can approximate a continuous function to any desired accuracy [31]. Moreover, ANN is basically a data-driven method [8]. Given a set of training samples, ANN can directly perform a mapping to foster the relationship between inputs and outputs. The relationship is stored by the network and can be used to make proper generalization.

In STLF, the output of ANN is the load value and the inputs are dependent variables, such as historical load values, temperature and calendar indicators. The process of input-output mapping is known as neural network learning, in which the weights and biases of the network are tuned to make the actual output move closer to the desired output. The learning process requires a number of samples, which are produced from historical data. Each sample is a pair consisting of an input vector

and a desired output vector. In the beginning, the weights and biases of ANN are randomly initialized. Then the weights and biases are adjusted to optimize the network performance, which is usually defined by a loss function such as mean square error (MSE). The learning process will end when a minimum of loss function is reached. After the learning process is completed, the ANN is available to forecast future load.

The application of ANNs to load forecasting was first published in the late 1980s. A thorough review of ANNs for STLF is available in [12], which examines a collection of journal papers published between 1991 and 1999. In this review, the approaches taken by each paper were compared. The key choices and procedures in designing an ANN based forecasting system were summarized, which provided good instructions for the researchers in this area. The design task can be roughly divided into four components: data pre-processing, designing, implementation and validation. On the other hand, this review also indicated two major weaknesses that make the performance of ANNs not very convincing. Firstly, most of the ANN models seemed to have been overtrained, which cannot achieve accurate out-of-sample forecasts. Secondly, in most cases, the numerical tests were not systematically performed, i.e., the models were not properly compared to the standard benchmarks.

In [32], a non-fully connected ANN was used to predict the hourly loads of an entire week. The training time was reduced as compared to the fully-connected network. In [33], the electrical load was expressed as the output of a dynamic system, in which the exogenous factors were

taken as the input variables. Recurrent ANNs were used to construct the empirical models for the dynamic system. In [34], it was found that the performance of ANN was system dependent, i.e., the optimal model for one utility was not necessarily appropriate for another one. In [35], an ANN method was proposed to forecast the load of a large power system. It was claimed that the ANNs, when grouped into different load patterns, provided a good load forecast. In [36], an adaptive learning algorithm with ANN was used for STLF. The effects of learning rate, momentum and other factors on the efficiency of the backpropagation momentum method were studied.

The application of ANNs to load forecasting is not limited to point forecast. In practice, ANNs can also generate prediction intervals and conditional probability densities [37, 38].

e) Expert Systems

An expert system is a computer system that simulates the problem-solving skills of a human expert [39]. It can be separated into two parts: knowledge base and inference engine. The knowledge base comprises a well-organized body of facts, rules and procedures in a bounded field of interest. The inference engine utilizes complex inferential reasoning to perform tasks. To design a good expert system, a human expert needs to work closely with the programmer, who studies the expert's knowledge and translates it into rules that a computer can understand.

Expert systems have found broad applications in STLF. In [40], a knowledge-based expert system was proposed for the short-term load

forecasting of the Taiwan power system. In the proposed expert system, eleven day types were identified based on the operator's knowledge and a five year dataset. Weather variables, such as temperature and relative humidity, were also considered. The proposed expert system performed better compared to the Box-Jenkins method. In [41], a site independent expert system based method was proposed for STLF. The electric load was modeled using a parameterized rule base and a parameter database. The load model, rules and parameters were designed using no specific knowledge about any particular site. The forecast results were expected to be improved if the knowledge about a particular site was included. In [42], a hybrid STLF method combining a fuzzy logic expert system and ANN was presented. The fuzzy logic module was employed to map the highly nonlinear relationship between the weather parameters and their impact on the daily load peak. Furthermore, the fuzzy logic output data, together with the load and weather data were used for ANN training.

f) Support Vector Machines

Support vector machine (SVM), pioneered by Vapnik, is a popular learning model for data classification and regression analysis [43]. The scheme of a SVM is to find a hyperplane as the decision surface in such a way that the separate classes are divided by a clear gap that is as large as possible [44]. For the linearly separable problems, SVM finds the optimal hyperplane which has the largest margin of separation. For the linearly inseparable problems, SVM uses a technique called *kernel trick* to efficiently map the original data into a higher dimensional space, in

which the data become separable. In general, the SVM technique offers the following advantages [45]:

1) SVM can be very resistant to overfitting and obtain a good out-of-sample generalization performance, even when the number of attributes is larger than the number of observations. The key to avoid overfitting is to appropriately select the regularization parameters.

2) The solution of SVM is a global one, because the optimization problem is convex. This implies that the issue of local minima will not appear in SVM modeling.

3) SVM can circumvent the curse of dimensionality. The reason is that SVM is an approximate implementation of a bound that is independent of the dimensionality of the feature space.

In [46], the method of SVM was used to perform STLF. The result indicated that SVM compared favorably against the AR model based on the same data. It was also remarked that the forecasting performance of SVM could be further boosted by increasing the training data. In [47], it was assumed that there was a strong correlation between the fluctuation part of load and the deviation of exogenous variables. SVM was used to model this relationship and produce the forecast of the fluctuation part. Then the output of SVM was added to the base load, which was offered by the time series model directly. In [48], an adaptive two-stage hybrid model for STLF was presented. In the first stage, a self-organized map network was applied to cluster the input data into several subsets. In the

second stage, a group of 24 SVMs was employed to fit the training data of each subset for the next day's load profile.

g) Other Alternatives

In addition to the above techniques, some other methods for STLF have also been reported. In [49], an optimal fuzzy inference system was proposed for load forecasting. The optimal system structure could grasp the nonlinear behavior of short-term loads with minimized model errors and number of membership functions. In [50], a composite load model containing three components: nominal load, type load and residual load was proposed for load forecasting. The nominal, type and residual load components were modeled using Kalman filter, exponential smoothing and AR model, respectively. In [51], a unique temperature match based optimization method was presented for STLF. This approach was based on a technique for detecting the uneven temperature match as well as a technique for reducing the forecast errors using the two error reduction procedures.

To improve the forecasting accuracy, some auxiliary methods have been combined with the artificial intelligence techniques to construct a hybrid forecast model. For instance, evolutionary algorithms served the ANNs to improve their local search ability for a possible better solution [52-55]. A phase-embedding technique was used in [56] to identify the dominant input variables that contribute to the forecasting performance. It can be seen that the development of hybrid forecasting models is one of the major directions in STLF.

2.3 Wind Power Forecasting

2.3.1 Overview

Wind power has become a promising alternative to fossil fuels for electric power generation, which grows at a significant rate around the world. By 2013, the installed commercial wind power in more than 90 countries and regions reached a total capacity of 318 GW, offering about 3% of global electricity [57]. It is estimated that wind power will increase to supply 12% of global electricity by 2020 [58]. Therefore, wind power is an important supplement to meet the fast growing electricity demand. However, due to the stochastic nature of the surrounding atmosphere of wind farm, the wind power generation shows a great variability. This variability results in high uncertainty and make the wind power integration more challenging. An effective solution is to build an accurate wind power forecasting (WPF) system for power grid management.

WPF estimates the future energy production by a number of wind turbines. WPF helps the utilities to carry out real-time grid operations, schedule spinning reserves and assess power quality [59]. Accurate forecast results can be used to develop highly-functional hour ahead or day ahead energy markets. Since wind farms are always built in remote areas, WPF is necessary for the companies to plan their transmission systems.

WPF is however an arduous task because of the intermittency of wind [60]. For example, wind turbines need wind to drive the blades. If there is no wind or the wind speed is too low, the turbines will not work. If the wind speed exceeds the rated value, the turbine must be turned off for security reasons [61]. The

wind speed depends highly on the local weather regime, which is quite unpredictable. The series of wind speed usually shows a chaotic behavior [62]. Unlike load series, wind speed has no apparent daily and weekly cycles. Therefore, the forecast accuracy of electric load is much better than that of wind power. For example, day ahead load forecasting can be achieved with an error of 2%–5%, whereas the error of WPF is around 10%–20% for the same forecast horizon.

WPF is a site-dependent system. The wind power production is affected by wind farm and terrain characteristics, such as temperature, air pressure, surface roughness and obstacles [11]. Hence, the existing experience of WPF cannot be directly applied to a new wind farm. It is necessary to adjust the forecast model with regard to the local circumstances.

Similar to load forecasting, WPF can be divided into four categories based on the forecast horizon: very short-term (few seconds to 30 minutes), short-term (30 minutes to 6 hours), medium-term (6 hours to 1 day) and long-term (1 day to 7 days) [60]. Different forecasts are made to meet different requirements. For instance, very short-term forecast is used for turbine control and load tracking, and long-term forecast is used for maintenance scheduling [11].

2.3.2 Wind Speed and Wind Power

Wind turbines convert kinetic energy from the wind into electricity. The power output of a wind turbine is considerably influenced by wind speed. The theoretical relationship linking wind power and wind speed is shown as [61]

$$P_w = \frac{1}{2}\rho A v^3 \tag{2.8}$$

where P_w is the wind power (W), A is the cross-sectional area (m^2), ρ is the air density (kg/m^3) and v is the wind speed. The cross-sectional area A is the swept area of the turbine blade. For a horizontal axis turbine, A is $\pi D^2/4$, where D is the blade diameter. It is noticed that the wind power-speed relationship is cubic. For example, doubling the wind speed will increase the power by eight-fold. A small error in wind speed forecast will cause a big bias in wind power forecast. Moreover, the wind power is proportional to the swept area of the turbine blade. For a horizontal axis turbine, the swept area A is $\pi D^2/4$, so the wind power is proportional to the square of the blade diameter. Doubling the diameter will add the power by four-fold. Thus, the capacity of wind turbine can be increased by making the blade longer. In practice, the cost of a wind turbine is approximately proportional to the diameter, but the power is proportional to the square of the diameter, so the wind turbines with longer blades are more cost effective [61]. In addition, the wind power is proportional to the air density, which is highly dependent on the atmosphere pressure as well as temperature.

A typical power curve of a wind turbine is given in Figure 2.6. The wind turbine does not work at very low wind speed. The cut-in speed is the minimum wind speed at which a wind turbine starts to rotate and yield electrical power. The cut-in speed is usually between 3 and 4 m/s. As the wind speed rises above the cut-in speed, the power generated by turbine increases rapidly, which can be seen from Figure 2.6. However, the power cannot always grow and will reach a limit that the turbine is capable of. The minimum speed at which a wind turbine generates the rated power is called the rated output speed (typically between 12 and 17 m/s). The turbine is fixed to deliver the rated power even the wind speed

is higher than the rated output speed. As the wind speed rises above the rated output speed, the force of wind continues to increase and there is a risk to break the turbine. The wind speed at which a wind turbine is shut down for avoiding possible damage is called the cut-out speed, which is around 25 m/s.

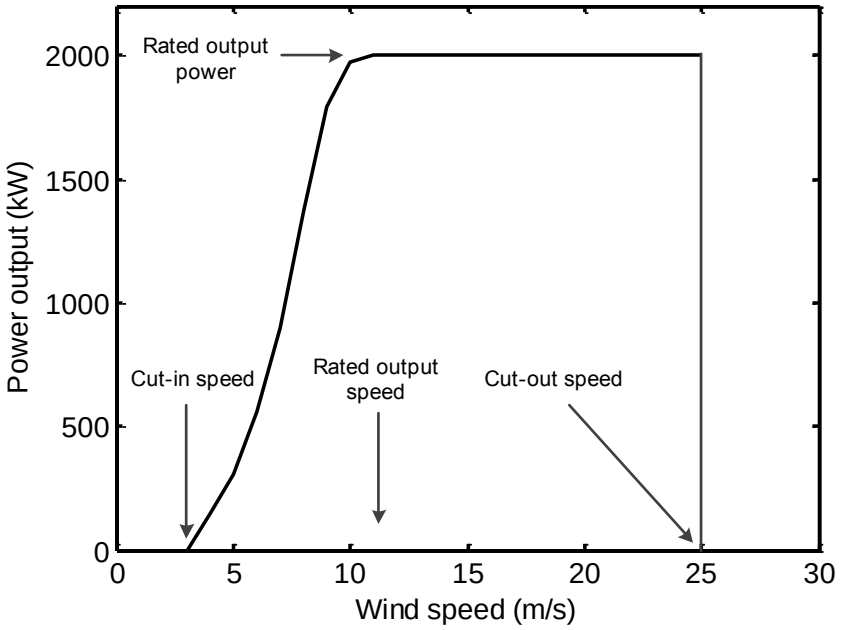


Figure 2.6 Power curve of Vestas V110-2.0 MW [63].

It is clear that the kinetic energy cannot be totally captured by the turbine when the wind passes through the blades. So there exists a maximum possible conversion efficiency from kinetic energy to electrical power. The maximum theoretical efficiency of a rotor is 59.3%, which is called the Betz efficiency or Betz's law [61]. It occurs when the wind turbine reduces the wind speed to one third of its original speed. In practice, the wind turbine can only reach 80% of the Betz efficiency at most. This implies that the wind turbine efficiency is in the range of about 45%–50% under the best operating conditions [61].

2.3.3 Wind Power Forecasting Methods

The penetration of wind energy has generated great interest in the research of WPF methods. In this section, an overview is presented to highlight the most common WPF techniques, rather than surveying all the existing models.

a) Persistence Method

The persistence method is the simplest way to forecast wind power production. It states that the wind power output in the near future is the same as its last measurement, i.e.

$$P_w(t+\Delta t) = P_w(t) \tag{2.9}$$

where P_w is the wind power at time t and Δt is the forecast horizon. The persistence method is based on the fact that there is a strong correlation within the wind power output data. It performs well for very short-term forecast but its error grows rapidly with the increasing forecast horizon. It is noted that any advanced method for WPF should first be compared against the benchmark of persistence method [60].

b) Physical Method

The physical method uses a detailed physical description of the on-site atmosphere to predict the wind power generation [11]. The physical considerations include terrain and meteorological variables, such as air pressure, temperature, humidity, obstacles, surface roughness and wind farm layout. The diagram of the physical method is given in Figure 2.7 [11].

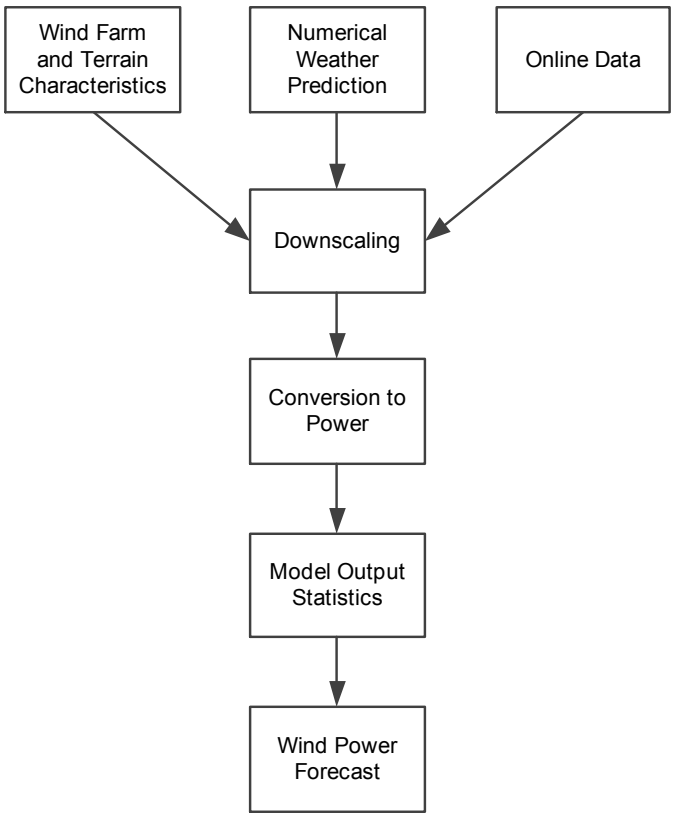


Figure 2.7 Diagram of the physical method.

According to Figure 2.7, the physical method can be separated into two phases. First, the numerical weather prediction (NWP) system tries to produce the best possible wind speed forecasts by solving a group of physical models. Second, the forecasted wind speeds are converted into wind power forecasts with the considerations of turbine hub height and siting, surrounding terrain and power curves of wind turbines, etc. The conversion process usually consists of two stages:

1) Downscaling: The wind speed predictions are scaled to the hub height of wind turbines using a set of meso- and micro-models.

2) Conversion: The scaled wind speeds are transformed to power outputs based on the power curves of wind turbines. Two kinds of power curves are available for selection: manual curve and measured curve. The first one is provided by the manufacturer, while the latter is derived from actual measurements. It is suggested that the measured power curve is better for WPF.

Many physical methods have been presented for WPF. An on-line automatic prediction system was developed to estimate the wind power output [64]. It contained many models, like NWP, HIRLAM, WAsP and PARK programs, and two MOS modules. In [65], a WPF system called *Previento* was proposed to forecast the power outputs up to 48 hours. It took advantage of the NWP results provided by the *Lokalmodell* model of the German Weather Service.

The physical method requires a large amount of measured data and good quality data. The complex physical models are solved using super computers. Even so, the computational cost is still very high. Thus, the physical method is usually used for medium- and long-term WPF (from 6 hours to 72 hours ahead) [60, 66].

c) Statistical Method

The statistical method aims to find a relationship between the wind power and a set of variables including historical wind power values and historical and predicted wind speed and direction values. The statistical method is simple to design and implement since it does not require any postulated model. In some cases, the statistical model can generate high

quality forecast results with historical wind power data only. Compared to the physical method, the computing burden of the statistical method is significantly reduced. It should be noted that the performance of the statistical method degrades rapidly when the look-ahead time increases. Hence, it is well-suited to perform very short-term and short-term WPF. The time series models like ARIMA and the artificial intelligence based models including ANNs, fuzzy logic and SVMs are examples of the statistical method. The fundamentals of these models have been stated in Section 2.2.4. Here, only some representative papers are reviewed to show the research directions of wind power forecasting.

In [67], a *fractional*-ARIMA model was employed to forecast wind speeds on the day ahead (24 h) and two-day ahead (48 h) horizons. The wind speed data were obtained from four stations in North Dakota. The forecasted wind speeds were converted to wind power predictions using the power curve. In [68], an improved time series method (ITSM) was proposed for WPF. The sub-series obtained from the wavelet transform were separately forecasted by ITSM and then combined to generate the aggregate forecasting. The simulation results revealed that the proposed method performed much better than the backpropagation network.

In [69], a new forecast engine composed of modified hybrid neural networks and enhanced particle swarm optimization was presented for WPF. Moreover, a feature selection component was integrated to select a compact set of input variables for the forecast model. In [70], a two-stage statistical model was proposed, which didn't use the NWP results.

In stage-I, the wind speed was forecasted by an adaptive wavelet neural network. In stage-II, a feedforward neural network was used to map the power-speed relationship, which transformed the forecasted wind speed to wind power prediction. In [71], a ridgelet neural network (RNN) was proposed as the main forecast engine, which used ridge functions as the activation functions of the hidden nodes. Moreover, a new differential evolution algorithm with novel crossover operator and selection process was presented to train the RNN. In [72], a detailed comparison study on the WPF application of three types of ANNs: adaptive linear element, backpropagation and radial basis function was presented. It was found that no single ANN model outperformed others universally with respect to all evaluation metrics. Moreover, the performance of a single ANN is highly dependent upon the data sources.

In [73], a *k*-nearest neighbors algorithm based model was proposed for WPF. The modeling procedure was made up of several steps: factor selection, raw data pretreatment, model evaluation and optimization. In [74], a fuzzy model was suggested to predict the wind speed and power at a wind park. The model was trained using a genetic algorithm based learning algorithm. The training data were not only obtained from the weather stations in the wind park, but also from the stations around the wind park. The autocorrelation and cross-correlation between local and remote wind speed time series were utilized to improve the forecasting efficiency. In [75], a new hybrid intelligent model based on the wavelet transform and fuzzy ARTMAP network was proposed for predicting the

power generation of a wind farm. The meteorological information such as wind speed and direction, and temperature were used. In [58], a new hybrid model with Bayesian clustering by dynamics and support vector regression was developed. The model was well-suited for capturing the dynamics of wind power/speed time series by using hybrid architecture, and could be easily modified for different wind farms. In [76], a hybrid complex-valued model was proposed using grey relational analysis and wind speed distribution features for very short-term WPF. The weight of each independent model was estimated according to different wind speed subsection and similar wind speed frequency. In [77], a Gaussian process (GP) combined with a NWP model was applied to forecast the wind power up to one day ahead. The wind speed data from NWP was corrected by a GP. Then the relationship between the power output and the corrected wind speed was modeled using a censored GP. In [78], a novel hybrid approach consisting of wavelet transform, particle swarm optimization and adaptive-network-based fuzzy inference system was proposed for short-term WPF in Portugal.

d) Ensemble Forecasting

Ensemble forecasting involves a number of different models which are allocated to solve the same problem [79]. The individual models in the ensemble typically have different initial conditions that can result in different forecast outputs. An ensemble model is expected to present a better performance than any of the individual models. The reason is that the diverse forecast errors of the individual members can cancel out in

the combining process [80].

Many ensemble forecasting methods have been proposed for WPF [81-83]. In [84], weather ensemble predictions were transformed to wind power density forecasts using the power curves of wind turbines. In [85], an ensemble model of 52 ANN sub-models and 5 GP sub-models was used for WPF. In this model, the NN sub-models were used to train the wind information and the GPs were used for providing the initial power level to the NN sub-models.

e) Probabilistic Forecasting

The forecasting models studied so far only provide a point forecast of the wind power output in the near future. Due to the intermittency of wind, the deterministic point forecast is unable to obtain high accuracy. For example, it is difficult to construct a precise point forecast model if the local weather regime is extremely chaotic. Therefore, probabilistic forecasting has attracted a lot of attention, which produces a prediction interval to assess the forecast uncertainty [86]. A prediction interval is an interval associated with a random variable yet to be observed, with a certain probability of the random variable lying within the interval. Probabilistic forecasting can give more meaningful insights for decision making, as compared to deterministic point forecast [87, 88].

In [89], a probabilistic WPF algorithm was proposed based on the quantile regression method. It made use of the forecast results obtained from the hybrid intelligent deterministic model, which was developed using wavelet transform and support vector machine. A novel extreme

learning machine based probabilistic forecasting method was proposed in [90]. In order to construct the prediction intervals, different bootstrap methods were carefully compared and analyzed to select the best one. In [91], a novel hybrid approach was proposed to directly formulate the optimal prediction intervals of wind power based on extreme learning machine and particle swarm optimization. The prediction intervals were created through the optimization of both the coverage probability and sharpness to ensure the quality. Moreover, the proposed model did not involve the distribution assumption of forecasting errors, which were yet needed in most existing methods.

2.4 Discussion

The conventional and state-of-the-art methodologies for short-term load and wind power forecasting are reviewed in this chapter. It is seen that there are many common factors between load forecasting and wind power forecasting. Some methods of load forecasting such as linear regression, time series and neural networks can also be applied to wind power forecasting. In order to make an accurate forecast, most of the papers focus on two aspects: data pre-processing and forecast engine designing. Data pre-processing may involve reducing the dimension of the input vector (so as to avoid the curse of dimensionality) or cleaning the data (by removing outliers or interpolating missing values). The purpose of data pre-processing is to make the forecasting process more efficient and manageable [12]. On the other hand, designing a proper forecast engine plays a key role in an energy forecasting problem.

Taking ANN as an example, the choices of network architecture, learning algorithm and number of hidden neurons would greatly affect the forecasting capability.

Chapter 3 Short-Term Load Forecasting Using Wavelet Transform and Evolutionary Extreme Learning Machine

3.1 Introduction

Short-term load forecasting (STLF) is always essential for power system operation and planning, which aims to estimate the load demand from one hour ahead to a week ahead. Among all the developed STLF methods, ANNs have received a high share of attention. A comprehensive review and evaluation of ANNs for STLF has been presented in [12].

In STLF, the most prevalent ANN architecture is the multilayer perceptron, which is usually trained by a gradient-based algorithm called backpropagation (BP). However, the conventional BP algorithm and its variants have two major shortcomings, which become the main bottleneck blocking the applications of ANNs. Firstly, the iterative learning process is time-consuming, possibly caused by the improper learning rate or the massive tunable network parameters (e.g. weights). Secondly, the performance of BP networks is not fully guaranteed. As gradient descent is performed on the error surface, the iterative learning process

might be trapped in one of the local minima. The gradient-based algorithms do not always reach the minimum training error. Furthermore, the network might be overtrained because there are too many parameters to be estimated from the limited training dataset. The network may have memorized the training samples but cannot produce good out-of-sample forecasts.

Recently, extreme learning machine (ELM) has been proposed to train the single-hidden layer feedforward neural networks (SLFNs), which can overcome the drawbacks faced by the gradient-based algorithms [92]. Unlike the popular understanding that all the parameters (e.g. weights and biases) need to be fine-tuned in the learning session, one may not necessarily adjust the input weights and the hidden layer biases [92]. In ELM, the input weights and hidden biases are initialized using a set of random numbers. The output weights of hidden layer are directly determined through a least squares method. In such a way, ELM completes its learning process. ELM has been successfully used in many applications, including electricity price and load forecasting [93-99].

In this chapter, an intelligent hybrid method is proposed for STLF based on the ELM. Two improvements are carried out to support the ELM for improved forecasting accuracy. Firstly, wavelet transform is able to produce an in-depth time and frequency representation by decomposing the signal into several sub-components with different frequencies [100-106]. This feature enables wavelet transform to conduct analysis on nonstationary signals [107]. The reason is the filtering effect of wavelet transform, which can extract the irregular information from the original signal. It is known that the load series is complex and exhibits several levels of seasonality [12]. We can use wavelet transform to decompose

the load series into a set of sub-components and then each sub-component is separately forecasted by a predictor. Following this scheme, we don't handle all the frequency components by a single forecaster but treat them differently. The nonstationarity problem of load series will be therefore eased. Secondly, it is found that ELM may yield unstable performance because of the random assignments of input weights and hidden biases [108]. In order to alleviate this problem, the modified artificial bee colony (MABC) algorithm is developed to look for the optimal input weights and hidden biases. MABC is a swarm-based optimization algorithm, which simulates the intelligent forging behavior of honey bee swarm [109]. MABC can be employed easily and does not require gradient information. Furthermore, MABC can probe the unknown regions in the solution space and look for the global best solution. This hybrid learning method can be named as ELM-MABC, which makes use of the merits of ELM and MABC.

The proposed method is tested on two datasets: ISO New England data and North American electric utility data. In order to evaluate the effectiveness, the proposed method is compared to other standard and state-of-the-art methods.

3.2 Methodology

3.2.1 Wavelet Transform

In pursuit of better accuracy, it is necessary to probe further into the load series. It is shown that the multiple frequency components in load series are the challenging parts in load forecasting [110]. A single forecaster cannot handle them appropriately and we can treat them differently with the help of wavelet

transform. Wavelet transform can be used to decompose a load profile into a set of constitutive components [111-113]. These constitutive components usually have better behaviors (e.g. more stable variance and no outliers) and therefore can be forecasted more accurately [110].

Wavelet transform makes use of two basic functions: scaling function $\varphi(t)$ and mother wavelet function $\psi(t)$. A family of functions can be derived from the scaling function $\varphi(t)$ and the mother wavelet $\psi(t)$ by

$$\varphi_{j,k}(t) = 2^{j/2}\varphi\left(2^j t - k\right) \tag{3.1}$$

$$\psi_{j,k}(t) = 2^{j/2}\psi\left(2^j t - k\right) \tag{3.2}$$

where j and k are integer numbers for scaling and translating [114]. The wavelet functions $\psi_{j,k}(t)$ and scaling functions $\varphi_{j,k}(t)$ are used for signal representation. Then a signal $S(t)$ can be expressed by

$$S(t) = \sum_k c_{j_0}(k)2^{j_0/2}\varphi\left(2^{j_0}t - k\right) + \sum_k \sum_{j=j_0}^{\infty} d_j(k)2^{j/2}\psi\left(2^j t - k\right) \tag{3.3}$$

where j_0 is the predefined scale, $c_{j0}(k)$ and $d_j(k)$ are the approximation and detail coefficients, respectively. It is seen that wavelet decomposition is performed to compute the above two sets of coefficients. The first term on the right of (3.3) presents a low resolution representation of $S(t)$ at the predefined scale j_0. For the second term, a higher resolution or a detail component is appended one after another from the predefined scale j_0 [115].

A demonstration of two-level decomposition for load series is given by

$$S(t) = A_1(t) + D_1(t) = A_2(t) + D_2(t) + D_1(t). \tag{3.4}$$

The load signal S is firstly decomposed to two components: A_1 and D_1. Then the approximation A_1 is further decomposed to two components: A_2 and D_2. The approximation A_2 reflects the general trend and offers a smooth form of the load signal S. The terms D_2 and D_1 depict the high frequency components in the load signal.

Three issues must be considered before using the wavelet transform: type of mother wavelet, number of decomposition levels and border effect. In order to choose the mother wavelet and number of decomposition levels, the trial and error method is employed, which has been used in many papers [103, 107]. Three popular wavelet families: Daubechies (db), Coiflets (coif) and Symlets (sym) [111] are studied for decomposing the load signal. The combinations of 12 mother wavelet functions (db2–db5, coif2–coif5 and sym2–sym5) and 3 decomposition levels (1–3) have been tested. It is found that the combination of coif4 and 2-level decomposition can yield the best forecasting performance. In addition, border distortion will appear when the wavelet transform is performed on finite-length signals, which would degrade the performance. The signal extension method in [107] is used in this chapter, which appends the previous measured values at the beginning of the load signal and forecasted values at the end of it.

3.2.2 Modified Artificial Bee Colony (MABC) Algorithm

The ABC algorithm, introduced by Karaboga, simulates the intelligent foraging behavior of honey bees [116]. The swarm in ABC is divided into three groups: employed bees, onlookers and scouts. The position of a food source represents a solution to the target problem while the nectar amount stands for

the fitness value of that solution. An employed bee may update its position in case of finding a new food source. If the fitness of the new source is higher than that of the old one, the employed bee chooses the new position over the old one. Otherwise, the old position is retained. After all the employed bees finish search missions, they share the information (i.e. positions and nectar amounts) of the food sources with the onlookers in the hive. An onlooker bee will choose a food source based on the associated probability value p_i, which is given by [117]

$$p_i = fit_i \Big/ \sum_{j=1}^{SN} fit_j \tag{3.5}$$

where fit_i is the fitness value of ith food source and SN is the number of food sources.

The basic ABC generates a new solution v_{ij} from the old one u_{ij} by [118]

$$v_{ij} = u_{ij} + \theta_{ij}\left(u_{ij} - u_{kj}\right) \tag{3.6}$$

where i and k are the solution indices and j is the dimension index. The index k has to be different from i and θ_{ij} is a uniformly random number within the range [-1, 1]. The old solution u_{ij} will be replaced by the new one v_{ij}, provided that v_{ij} has a better fitness value.

If a food source cannot be improved for many cycles, it will be abandoned. The number of cycles for abandonment is called *limit*, which is a crucial control parameter in ABC. The employed bee related to the abandoned source becomes a scout. The scout discovers the new food position by [109]

$$u_{ij} = u_{\min,j} + rand(0,1)(u_{\max,j} - u_{\min,j}) \tag{3.7}$$

where $u_{\min,j}$ and $u_{\max,j}$ are the lower and upper bounds for the dimension j,

respectively. The random number in (3.7) follows the uniform distribution.

It has been pointed out that the search equation given by (3.6) is good at exploration but poor at exploitation [119]. To balance these two capabilities and improve the convergence performance, a modified search equation is proposed as follows:

$$v_{ij} = w \cdot u_{best,j} + \theta_{ij}(u_{best,j} - u_{ij}) \tag{3.8}$$

where u_{best} is the best solution in current population, and w is the inertia weight. The search equation (3.8) uses the information of the best solution to direct the movement of swarm. The new solution is driven towards the best solution of the previous cycle. The coefficient w controls the impact from the best solution u_{best}. A large weight encourages the global exploration, while a small one speeds up the convergence to optima. In this chapter, the inertia weight w is chosen to be 0.1. Hence, the modified equation can improve the exploitation capability and accelerate the convergence speed. The search process of MABC will end if a stop criterion is satisfied. Normally, a maximum cycle number (*MCN*) is used to terminate the algorithm.

3.2.3 Evolutionary Extreme Learning Machine

a) Basic ELM

ELM is an emerging learning algorithm for SLFN, which chooses the input weights and hidden biases randomly and decides the output weights through a simple matrix inversion [120]. The main concept of ELM is that the learning problem is turned into a least squares problem, which can be easily solved in much less time.

Given a set of N training samples ($\boldsymbol{x}_i$, $\boldsymbol{d}_i$), where $\boldsymbol{x}_i$ is the ith input vector and $\boldsymbol{d}_i$ is the ith target output vector, the SLFN is modeled by

$$\sum_{j=1}^{n} \beta_j g\left(\boldsymbol{w}_j \cdot \boldsymbol{x}_i + b_j\right) = \boldsymbol{o}_i, \quad i = 1, \ldots, N \tag{3.9}$$

where n is the number of hidden nodes, $g(\boldsymbol{x})$ is the activation function, $\boldsymbol{w}_j$ is the input weight vector, b_j is the hidden bias vector, β_j is the output weight vector and $\boldsymbol{o}_i$ is the actual network output.

If ELM fits all the training samples ($\boldsymbol{x}_i$, $\boldsymbol{d}_i$) with zero error, it can be said that there exist β_j, $\boldsymbol{w}_j$ and b_j such that

$$\sum_{j=1}^{n} \beta_j g\left(\boldsymbol{w}_j \cdot \boldsymbol{x}_i + b_j\right) = \boldsymbol{d}_i, \quad i = 1, \ldots, N. \tag{3.10}$$

Equation (3.10) can be re-written in a matrix form by $\mathbf{H}\boldsymbol{\beta}=\boldsymbol{d}$, where $\boldsymbol{\beta}=[\,\beta_1, \ldots, \beta_n]^{\mathrm{T}}$, $\boldsymbol{d}=[\,\boldsymbol{d}_1, \ldots, \boldsymbol{d}_N]^{\mathrm{T}}$, and $\mathbf{H}$ is called the hidden layer output matrix and given by

$$\mathbf{H} = \begin{bmatrix} g\left(\boldsymbol{w}_1 \cdot \boldsymbol{x}_1 + b_1\right) & \cdots & g\left(\boldsymbol{w}_n \cdot \boldsymbol{x}_1 + b_n\right) \\ \vdots & \ddots & \vdots \\ g\left(\boldsymbol{w}_1 \cdot \boldsymbol{x}_N + b_1\right) & \cdots & g\left(\boldsymbol{w}_n \cdot \boldsymbol{x}_N + b_n\right) \end{bmatrix}_{N \times n}. \tag{3.11}$$

In practice, ELM cannot obtain the perfect zero error because the number of hidden nodes n is usually less than the number of training samples N. To train the network, the input weights $\boldsymbol{w}_j$ and hidden biases b_j are initialized using a set of uniformly distributed random numbers. For settled $\boldsymbol{w}_j$ and b_j, the hidden layer output matrix $\mathbf{H}$ can be computed and the SLFN becomes an over-determined linear system. So far, only the output weights $\boldsymbol{\beta}$ is unknown in the SLFN. Therefore, the objective

of learning is converted to find a least squares solution $\boldsymbol{\beta}^*$ to satisfy the linear system $\mathbf{H}\boldsymbol{\beta}=\boldsymbol{d}$. In ELM, a special solution is given by $\boldsymbol{\beta}^*= \mathbf{H}^\dagger\boldsymbol{d}$, where $\mathbf{H}^\dagger$ is the Moore-Penrose (MP) inverse of the hidden layer output matrix $\mathbf{H}$. It is advised that the singular value decomposition method is well-suited to compute the MP inverse of $\mathbf{H}$ in all cases [121].

ELM is able to produce good generalization performance because both the minimum training error and the smallest norm of weights can be directly obtained [92]. The special solution $\boldsymbol{\beta}^*$ is one of the least squares solutions of the linear system $\mathbf{H}\boldsymbol{\beta}=\boldsymbol{d}$, which implies that ELM can reach the minimum error and does not get trapped by local minima [120]. Moreover, $\boldsymbol{\beta}^*$ has the smallest norm among all the least squares solutions of $\mathbf{H}\boldsymbol{\beta}=\boldsymbol{d}$ **[121]**. It is noted in [122] that the smaller the weights are, the better generalization performance the network tends to have. In addition, ELM can get rid of many trivial problems faced by traditional learning methods, such as local minima, stopping criteria and learning rate [123]. It can be seen that ELM is a one-step learning algorithm, and therefore does not need stopping criteria. Moreover, the calculation of output weights does not involve the learning rate.

b) Evolutionary ELM

It is observed that the performance of ELM depends highly on the chosen set of input weights and hidden biases. ELM may have worse performance in case of non-optimal parameters. In this chapter, the proposed MABC algorithm is employed to find the optimal set of input weights and hidden biases for ELM.

Firstly, the initial population is generated and each candidate solution $\boldsymbol{u}_i$ consists of a set of input weights and hidden biases by

$$\begin{aligned}\boldsymbol{u}_i = [&w_{11}, w_{12}, ..., w_{1n}, w_{21}, w_{22}, ..., w_{2n}, ..., \\ &w_{m1}, w_{m2}, ..., w_{mn}, b_1, b_2, ..., b_n]\end{aligned} \tag{3.12}$$

where n is the number of hidden nodes and m is the number of input nodes. All the variables in the individuals are within the range [-1, 1]. Secondly, for each individual, the output weights are obtained through calculating the MP inverse. In this chapter, the root mean square error (RMSE) is chosen as the fitness function, which is given by

$$Fitness = \sqrt{\frac{1}{N}\|\boldsymbol{d} - \mathbf{H}\boldsymbol{\beta}\|^2} \tag{3.13}$$

The symbol $\|\cdot\|$ stands for the L^2 vector norm. Thirdly, the population is subjected to the search process of MABC. The optimal input weights and hidden biases are obtained until MABC completes *MCN* cycles.

The hybrid learning algorithm can take advantage of the merits of ELM and MABC. Firstly, MABC is a global search method with strong exploitation capability, which allows the learning algorithm to avoid the local minima and reach to the global minimum. Moreover, the optimal parameters from MABC guarantee that ELM has a small training error. Secondly, it should be noted that in the conventional neuro-evolution methods, all the network weights (i.e. input weights, hidden biases and output weights) are tuned by the evolutionary algorithm [100]. However, in ELM-MABC, only parts of weights (i.e. input weights and hidden biases) are adjusted by MABC. The output weights are not

determined by MABC but the least squares method. This difference can result in many advantages in training the network. The iterative minimization is performed over the set of input weights and hidden biases instead of all weight parameters. The learning process will be accelerated because fewer parameters are estimated. Furthermore, since the output weights are calculated by a least squares method at each iteration, the training error is always at a global minimum with respect to the output weights [124]. The robustness of the training process is highly improved.

The procedure of ELM-MABC, shown in Figure 3.1, is as follows:

1) Generate the initial population randomly. Each individual (i.e. candidate solution) in the population consists of a set of input weights and hidden biases.
2) For each individual, calculate the matrix **H** and the output weights $\boldsymbol{\beta}$.
3) Evaluate the fitness of each individual and start the search process.
4) Repeat the search process for *MCN* cycles.
5) Output the best solution as the optimal set of input weights and hidden biases.

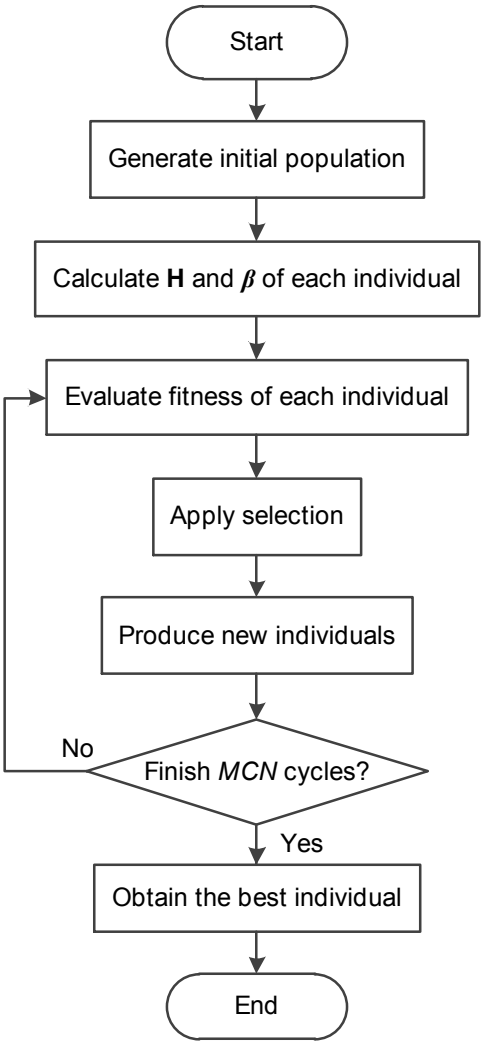


Figure 3.1 Flowchart of ELM-MABC.

3.2.4 Proposed Forecast Model

The structure of the proposed STLF model is shown in Figure 3.2. The step by step procedure can be summarized as follows:

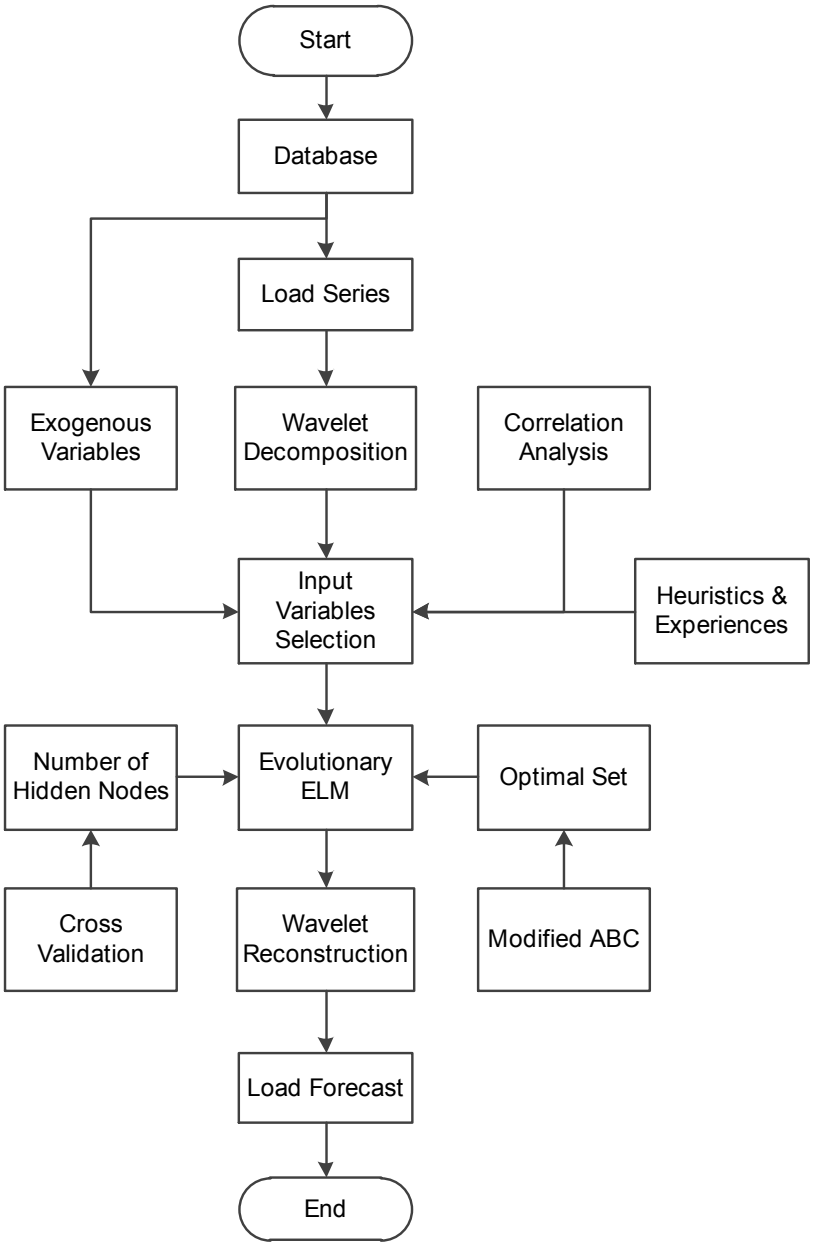


Figure 3.2 Structure of the proposed STLF model.

1) Extract the required data and divide them into load series and exogenous variables. It is noted that the resolution of data is one hour, unless otherwise stated.
2) Use the mother wavelet to break up the load series into three sub-series: A_2, D_2 and D_1, as discussed in Section 3.2.1.

3) Select the input variables for each sub-series. The details of input variables selection are presented in Section 3.3.1. Test different sets of input variables if necessary.
4) Determine the number of hidden nodes for each ELM. The number of hidden nodes is essential to the forecaster, but there is little theoretical basis for its selection [12]. In this chapter, the trial and error method is used. The number of hidden nodes is gradually increased by an interval of 5 from 10 to 100. The one that gives the best prediction performance on the validation set is selected.
5) Produce the optimal input weights and hidden biases using MABC. In MABC, the search process will terminate when finish *MCN* cycles. The number *MCN* is selected by the trial and error method. The one giving the best performance on validation dataset is used.
6) Evaluate the model on the validation dataset. If the prediction accuracy is not satisfactory, repeat steps 3) to 6). Otherwise, go to step 7).
7) Deploy the obtained model to forecast future load.

3.3 Results

3.3.1 Input Variables Selection

Input variables selection is a very important step of load forecasting. There are many variables that can be used in STLF. Generally speaking, more input variables may provide more accurate results. However, excessive variables are prone to cause many problems, such as prolonged training process, unnecessary storage space and the curse of dimensionality [125]. Therefore, a compact set of

input variables is selected for the predictor. Suppose the forecast hour is time τ, the following candidate variables are considered:

1) Historical load: Correlation analysis is used to select the most relevant historical load values as the load inputs. The 200-hour load data prior to the time τ is considered for selection.
2) Temperature: In most situations, temperature is the key factor to drive the variations of load consumption. The temperature values at time τ, τ-1, τ-2 and τ-24 are used as the temperature inputs in our model.
3) Day of the week: The numbers from 1 to 7 are used to mark the day of the week (e.g. 1 is used for Monday and 7 for Sunday).
4) Hour of the day: The load series usually exhibits a daily pattern and it is necessary to pass this information to the forecasting model. This can be achieved by defining two additional variables to codify the hour of the day. Two variables: $H_a=sin(2\pi h/24)$ and $H_b=cos(2\pi h/24)$ are included in our model, where h is the hour in a day (0, 1,…, 23) [56].
5) Weekend index: The numbers 1 and 0 are used to mark weekdays and weekends: 1 for a weekend and 0 for a weekday. All the holidays are regarded as weekends.

3.3.2 Case Studies

In this section, the proposed method is examined using the actual load and temperature data. The proposed method is also compared with other methods based on two publicly available datasets: ISO New England data [16] and North American electric utility data [126]. The two utilities are different in size, usage pattern of electricity and weather conditions. All simulations were conducted in

Matlab on a computer with a 2.66-GHz CPU and 3.25-GB memory. In order to evaluate the forecast performance, two error metrics: mean absolute percentage error (MAPE) and mean absolute error (MAE) were used. They are defined by

$$\begin{aligned} \text{MAPE} &= \frac{1}{M}\sum_{i=1}^{M}\left|\frac{A_i - F_i}{A_i}\right| \times 100\% \\ \text{MAE} &= \frac{1}{M}\sum_{i=1}^{M}\left|A_i - F_i\right|. \end{aligned} \tag{3.14}$$

where M is the length of data, A_i is the actual value and F_i is the forecast value.

a) Case 1

This case compares the convergence performance between the conventional neuro-evolution method (NN-MABC) and ELM-MABC. As mentioned in Section 3.2.3, in NN-MABC, all the weights are fine-tuned by the evolutionary algorithm (MABC is used as the evolutionary algorithm for fair comparison). In case of ELM-MABC, only the input weights and hidden biases are fine-tuned by MABC. For each learning method, 20 trials have been conducted and the average convergence performance has been reported. The maximum cycle number is set to be 500 for both learning methods. Figure 3.3 illustrates the convergence performance of NN-MABC and ELM-MABC, with different number of hidden nodes. The training data used are actual measurements from ISO New England.

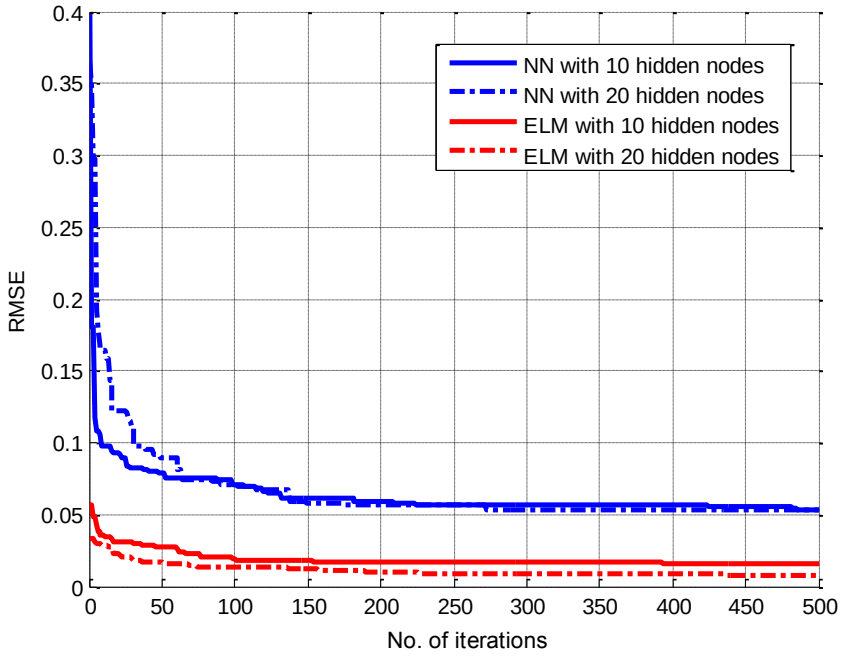


Figure 3.3 Convergence curves of NN-MABC and ELM-MABC.

It is observed that the convergence performance of ELM-MABC is significantly better than that of NN-MABC. After finishing 500 steps, the error of ELM-MABC is much smaller than that of NN-MABC. Moreover, the error curve of ELM-MABC is also far below its rival all the time. For the 10-node case, the error of NN-MABC at iteration 500 (0.0534) is slightly smaller than that of ELM-MABC at iteration 1 (0.0573). The gap of performance is wider in terms of the 20-node case. The error of ELM-MABC at iteration 1 (0.0330) is even lower than that of its rival at iteration 500 (0.0535).

The improvement in training SLFN benefits from the special learning mechanism of ELM. ELM-MABC converges faster than NN-MABC because only the input weights and hidden biases are estimated.

The output weights of the hidden layer were obtained by a least squares method. This implies that the training error of ELM-MABC is always at a global minimum with respect to the output weights. Therefore, ELM-MABC has a much better learning accuracy than NN-MABC, as shown in Figure 3.3.

b) Case 2

In this case, the proposed method was used to perform both 1-hour and 24-hour ahead load forecasting. The hourly load and temperature data were collected from ISO New England. The data from November 2009 to December 2010 were used in the simulations. In order to test the proposed method, for each month in 2010, the third week was selected as the testing week. The one week prior to the testing week was set to be the validation week. The five weeks before the validation week were used as the training weeks. Note that the training data will have a part of data from the previous month. The parameters of the forecast model were adjusted on the validation data. Since the testing period is only one week, the forecast model will not be retrained. In the simulations, actual temperatures were used as the forecasted values. In order to evaluate the effect of weather forecasting error, the forecasted temperatures were manually simulated. The Gaussian noise of zero mean and standard deviation of 0.6°C was added to the actual temperature data, as advised in [107].

The 1-hour and 24-hour ahead forecast results are shown in Table 3.1. It is obvious that the proposed method can yield satisfactory results

with two different time horizons. The average error is 0.5554 for 1-hour ahead case and 1.59 for 24-hour ahead case, respectively. The errors of 1-hour ahead case are much smaller than those of 24-hour ahead case. Moreover, the proposed method is able to produce encouraging results with simulated temperatures. With Gaussian noise, the average forecast error increases only 5.8% and 10.1% for 1-hour and 24-hour ahead forecasts, respectively.

Table 3.1 MAPE results (%) for 1-hour and 24-hour ahead forecasting

	Actual temperature		Noisy temperature	
	1-hour	24-hour	1-hour	24-hour
Jan	0.4330	1.43	0.4411	1.60
Feb	0.5543	1.27	0.5653	1.40
Mar	0.8631	1.53	0.9162	1.64
Apr	0.4721	1.65	0.4925	1.68
May	0.5025	1.30	0.5140	1.32
Jun	0.5435	1.56	0.5723	1.72
Jul	0.5812	1.73	0.6169	1.96
Aug	0.4529	2.59	0.5511	2.79
Sep	0.7038	1.39	0.7566	1.78
Oct	0.6141	1.60	0.6232	1.68
Nov	0.4770	1.58	0.5052	1.62
Dec	0.4678	1.48	0.4986	1.78
Mean	0.5554	1.59	0.5877	1.75

c) Case 3

This case inspects the influence of wavelet transform and MABC algorithm on the forecasting performance. The proposed method (WT-ELM-MABC) is compared with other three methods: ELM, ELM with wavelet transform (WT-ELM) and ELM with MABC algorithm (ELM-MABC). All the ELMs have only one output neuron. The sigmoid and linear functions are used in the hidden and output layers, respectively. The testing data are identical with those in Case 2. The results for 1-

hour ahead load forecasting are tabulated in Table 3.2. The findings are summarized as follows:

Table 3.2 MAPE results (%) for the models in Case 3

	ELM	WT-ELM	ELM-MABC	WT-ELM-MABC
Jan	0.6249	0.4471	0.6094	0.4330
Feb	0.5975	0.5854	0.5759	0.5543
Mar	1.2251	1.0302	0.8647	0.8631
Apr	0.5854	0.5432	0.5302	0.4721
May	0.6227	0.5421	0.5348	0.5025
Jun	0.8256	0.6755	0.7038	0.5435
Jul	0.8385	0.8006	0.6763	0.5812
Aug	0.7094	0.5327	0.6704	0.4529
Sep	0.9216	0.7806	0.8404	0.7038
Oct	0.7527	0.6549	0.7410	0.6141
Nov	0.5669	0.5048	0.5266	0.4770
Dec	0.7898	0.5349	0.6538	0.4678
Mean	0.7500	0.6360	0.6606	0.5554

1) It can be observed that the performance is greatly improved if wavelet transform is used. With the help of wavelet transform, WT-ELM has obtained an improvement of 15.2% with respect to ELM. Compared with ELM-MABC, WT-ELM-MABC has also experienced an increase of 15.9% in forecast accuracy.

2) The results of Table 3.2 indicate that the MABC algorithm is an effective tool to improve the forecast accuracy. Comparing WT-ELM with WT-ELM-MABC, the forecast error is reduced by 12.7% when the input weights and hidden biases are pre-optimized. For ELM and ELM-MABC, the accuracy is 13.5% worse if MABC is not used in the model.

3) It can be seen from the results that WT-ELM-MABC presents much better accuracy as compared to the other three methods.

On an average, the enhancements of WT-ELM-MABC are 25.9%, 12.7% and 15.9% respectively with regard to the previous three methods.

4) It is noteworthy to draw attention to the computational time of the above four methods. The first two methods ELM and WT-ELM only take a few seconds to complete the training and testing process, because ELM has no iterative steps. The time of the latter two methods becomes much longer since MABC is employed to optimize ELM. To accomplish the task, ELM-MABC and WT-ELM-MABC spend about two and five minutes, respectively.

d) Case 4

This case compares the proposed method to the ISO-NE method in [127] and the wavelet neural networks (WNN) method in [128] on the ISO New England data. As shown in [128], the load data may include spikes due to the possible malfunction of devices. These spikes do not reflect the actual load values and should be removed before forecasting. The spikes are classified into “micro spikes” and “macro spikes” based on their widths. The idea for filtering the micro spikes is the use of a zero phase filter to obtain the smoothed load. For macro spikes, the idea is to detect a pair of edges, and fix the loads between the two edges with linear interpolation values. After spike filtering, the load data with other inputs like time indicators are sent to the wavelet neural networks. The forecast range for comparison is from July 1, 2008 to July 31,

2008. As the testing period is longer than one week, the forecast model will be weekly updated to pursue better accuracy. Table 3.3 shows the 1-hour ahead forecasting results of the three methods. The results of ISO-NE and WNN methods are extracted from [128]. On the given testing data, the proposed method outperforms the WNN method, about 8.2% better in MAPE and 10.9% better in MAE. Compared to the ISO-NE method, the proposed method shows significant improvements in both metrics.

Table 3.3 MAPE and MAE results for the models in Case 4

	MAPE (%)	MAE (MW)
ISO-NE	0.81	138.33
WNN	0.49	83.54
Proposed	0.45	74.41

e) Case 5

In this case, a comparison is performed between the proposed method and the standard neural network (NN) method and the similar day-based wavelet neural network (SIWNN) in [129]. The SIWNN method selected similar day load as the input data based on correlation analysis and used wavelet neural networks to capture the load features at low and high frequencies. The training period is from March 2003 to December 2005. The proposed method is used to predict the hourly load data from January 1, 2006 to December 31, 2006. Only 24-hour ahead forecasting is considered. The forecast results of the three methods are shown in Table 3.4. It is clear that the proposed method produces the best forecast results. More precisely, the proposed method

is 27.1% and 13.5% better than the standard NN and SIWNN methods, respectively.

Table 3.4 MAPE results (%) for the models in Case 5

	Standard NN	SIWNN	Proposed
Jan	2.01	1.60	1.52
Feb	1.50	1.43	1.28
Mar	1.55	1.47	1.37
Apr	1.51	1.26	1.05
May	1.69	1.61	1.23
Jun	2.30	1.79	1.54
Jul	3.72	2.70	2.07
Aug	3.33	2.62	2.06
Sep	1.60	1.48	1.41
Oct	1.52	1.38	1.23
Nov	1.73	1.39	1.33
Dec	1.91	1.75	1.65
Mean	2.03	1.71	1.48

f) Case 6

This case compares the proposed method to four other methods on the North American electric utility data [107, 110, 130, 131]. In [110], a hybrid forecast method composed of wavelet transform, neural network and evolutionary algorithm was proposed for STLF. Specifically, a two-step correlation analysis was integrated to select the most informative input variables. In [107], to overcome the border distortion problem, a novel load signal extension scheme was proposed, which is also used in our model. Each component from the decomposition was forecasted separately. In [130], echo state network (ESN) was utilized as a stand-alone forecaster to deal with the STLF problem. No lagged load and temperature input variables were involved in the model because of the special property of ESN. In [131], a parallel model using 24 SVMs was proposed to conduct the day-ahead load forecast. The parameters of

SVMs were optimized by the particle swarm pattern search.

The loads and temperatures from January 1, 1988 to October 12, 1992 were used to run experiments. The hourly loads for the two-year period prior to October 12, 1992 were forecasted. Both the hour ahead and day ahead load forecasts were considered. Moreover, the effect of noisy temperature was also studied. Table 3.5 compares the results of the proposed method with the other four methods proposed in [107, 110, 130, 131]. It can be seen that the proposed method can produce superior results to the other methods in all testing cases. The 24-hour ahead forecast results of the proposed method are shown in Figure 3.4.

Table 3.5 MAPE results (%) for North American electric utility

	Actual temperature		Noisy temperature	
	1-hour	24-hour	1-hour	24-hour
[107]	1.10	2.64	1.11	2.82
[110]	0.99	2.04	n.a.	n.a.
[130]	1.14	2.37	1.21	2.53
[131]	0.72	1.99	0.73	2.03
Proposed	0.67	1.87	0.69	1.95

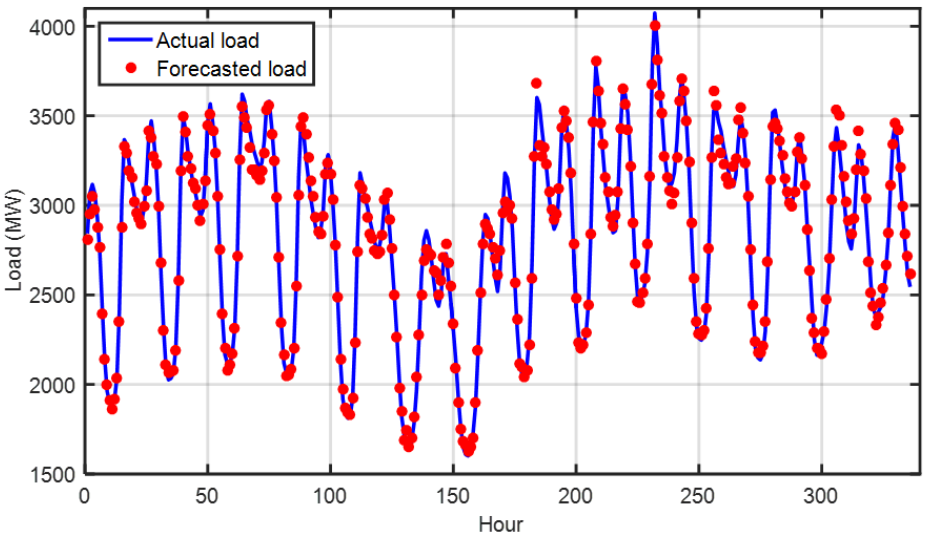


Figure 3.4 Forecast results of 14 days using actual temperature.

g) Case 7

In this case, the effect of temperature forecasting error on STLF was further studied using the North American electric utility data. In order to cover a large scale of temperature errors, a set of Gaussian noises with different means and standard deviations were considered in 1-hour ahead forecasting. The MAPE result (0.67) obtained with actual temperatures serves as the reference. Using different noises, the MAPE increments with respect to the reference are shown in Figure 3.5. It is seen that the noises with large means or deviations will bring large load forecasting errors. For example, the forecasting error has a 12.2% increase when the noise is with mean 3 and standard deviation 3.6. The forecast results with zero-mean Gaussian noises are presented in Table 3.6. The associated temperature error ranges are also provided in the table. It is clearly observed that the load forecasting error grows much slower than the temperature error. In that context, it can be said that the proposed method is relatively robust with respect to temperature errors. The forecasting error only climbs 9.61% when the temperature error varies in the largest interval [-14.1 °C, 15.2 °C].

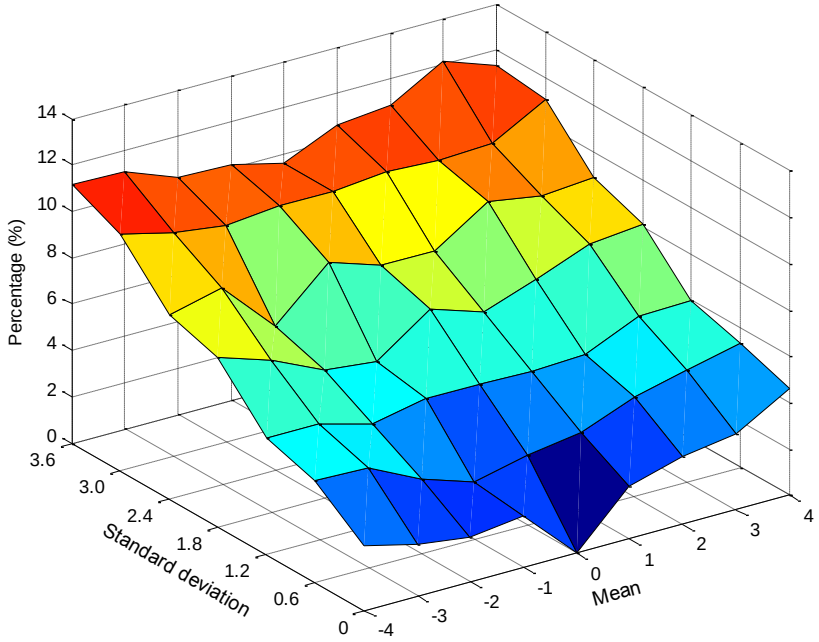


Figure 3.5 MAPE increments due to different Gaussian noises: means={-4, -3, -2, -1, 0, 1, 2, 3, 4} and standard deviations={0, 0.6, 1.2, 1.8, 2.4, 3.0}.

Table 3.6 1-hour ahead forecast results with zero-mean Gaussian noises

Standard deviation	Simulated temperature error range (℃)	Load forecasting error increment
0	n.a.	0
0.6	[-2.3, 2.1]	2.99%
1.2	[-4.5, 4.6]	4.88%
1.8	[-7.2, 6.7]	6.91%
2.4	[-9.8, 9.3]	7.58%
3.0	[-10.7, 11.9]	9.55%
3.6	[-14.1, 15.2]	9.61%

3.4 Conclusion

This chapter proposes a novel hybrid model for STLF based on the ELM. Two auxiliary techniques are developed to assist the ELM. Wavelet transform is used to decompose the load series into a set of different frequency components, which are more predictable. Moreover, a modified ABC algorithm is proposed

to choose the optimal input weights and hidden biases for ELM. The ELM-MABC algorithm has better convergence performance than the conventional neuro-evolution method, leading to an improvement in forecasting accuracy.

To confirm the effectiveness, the proposed hybrid method has been tested using the actual data from two public datasets. The simulation results reveal that the proposed method can produce excellent forecasting results over other well-established methods. There are several factors that contribute to the improved forecasting accuracy, such as the special learning mechanism of ELM, the integration of wavelet transform, the optimal parameters from MABC and the appropriate selection of input variables.

The proposed method presents many advantages in STLF. Firstly, wavelet transform decomposes the load signal into several sub-components, which have better behavior and can be forecasted more accurately. Following this scheme, the difficulty induced by the nonstationarity will be alleviated. Secondly, it has strong robustness in terms of large temperature forecasting errors. Thirdly, it can produce accurate load predictions for electric utilities with different sizes and weather conditions. In addition, for the above case studies, the maximum training time of the proposed method is 38 minutes. In contrast, the testing time of the proposed method only takes several seconds, which is negligible.

Chapter 4 A Novel Wavelet-Based Ensemble Method for Short-Term Load Forecasting with Hybrid Neural Networks and Feature Selection

4.1 Introduction

Load forecasting has a major impact on many aspects of power industry, such as energy market analysis, economic dispatch and security assessment. It is suggested that an increase of a few percentage points in forecasting accuracy will save millions of dollars [132]. The importance of load forecasting inspires the continuous development of forecasting techniques. In recent times, artificial intelligence based methods, especially the ANNs [12, 20, 34, 36, 104, 133-141], have been extensively utilized for load forecasting.

ELM as a special class of ANN has been effectively used in solving power system problems, including price and load forecasting [93-95]. Compared to the conventional ANNs, ELM can circumvent many trivial problems such as local minima, heavy computing burden and improper learning rate [123]. However, there is always a lot of room for improvements. As pointed out in Chapter 3, the random assignments for input weights and hidden biases may bring non-optimal

parameters, leading to unreliable performance. Moreover, the overtraining issue may arise if there are too many parameters estimated from the training data. To alleviate the above problems, this chapter proposes a neural network ensemble model for STLF.

A neural network ensemble model involves many individual ELM-based forecasters, which are allocated for the same load forecasting problem. It has been shown that ensemble STLF models have advantages with respect to single models in terms of accuracy and robustness [142-144]. The improvements can be attributed to the fact that the diverse errors of individual forecasters can cancel out in the aggregating process [80].

The wavelet transform has been successfully used for STLF by decomposing the load data into a set of sub-components [100]. However, it can play a much greater role in the modeling process. This chapter proposes a novel ensemble scheme, which uses wavelet transform to generate the collection of individual predictors. It is known that two key wavelet parameters are required in the transform: type of mother wavelet and number of decomposition levels. In the proposed ensemble model, the individual predictors are obtained using different combinations of the mother wavelet and the number of decomposition levels. The rationale lies in that different wavelet settings would create different input variables for the ELM-based predictors, which can promote the diversity of the network ensemble.

In ensemble forecasting, simple averaging is widely used to aggregate the individual forecasts to form an ensemble forecast [95, 142], which presumes that each individual has the same contribution. But in practice, some individual

forecasts in the ensemble are actually more accurate than others. The ensemble output should be a weighted average of the individual forecasts. In this chapter, the weight factors are obtained using the partial least squares regression (PLSR) method.

In the development of an STLF model, determining a suitable set of input variables from the raw data has a big impact on the forecasting performance. Generally speaking, more input variables carry more discriminating power, but in practice, excessive variables would bring many drawbacks [125]. The reason is that some variables are irrelevant or redundant to the model, which could confuse the learning algorithm and result in poor results. This chapter proposes a conditional mutual information based feature selection (CMIFS) technique to select a small set of informative input variables that are beneficial to STLF.

In order to improve the learning accuracy of SLFN, this chapter proposes a hybrid learning algorithm combining ELM and Levenberg-Marquardt (LM) method [145]. As shown in Section 3.2.3, ELM is a one-step learning algorithm based on the least squares method. In contrast, LM is an iterative learning algorithm, which adjusts the network parameters using the gradient information of network errors [8]. The proposed hybrid algorithm can take advantage of the merits of both ELM and LM. In the learning process, the input weights and hidden biases are fine-tuned using LM, while the output weights are computed by the least squares method from ELM. This hybrid learning algorithm can be named as ELM-LM.

This chapter proposes a novel ensemble model for STLF, in which wavelet transform, hybrid neural networks, feature selection and partial least squares

regression are involved. In order to confirm the effectiveness and feasibility, the proposed method is tested using actual data from two electric utilities: ISO New England and North American electric utility.

4.2 Methodology

4.2.1 Hybrid Neural Networks

ELM has been successfully utilized in many applications; however, it also has its own limitations. As pointed out in Chapter 3, the training error of ELM depends highly on the initial input weights and hidden biases. In case of inferior input weights and hidden biases, ELM would have poor learning accuracy. In order to boost the learning accuracy, a hybrid algorithm combining ELM and Levenberg-Marquardt algorithm is proposed for neural network training. In this iterative hybrid algorithm, the neural network parameters are divided and tuned separately: the input weights and hidden biases are updated by the Levenberg-Marquardt algorithm while the output weights are directly calculated using the least squares method from ELM.

Given a set of training samples ($\boldsymbol{x}_i$, $\boldsymbol{d}_i$), the goal of training an SLFN is to look for a specific set of network parameters $\boldsymbol{w}$, $\boldsymbol{b}$ and $\boldsymbol{\beta}$ to optimize the network performance in the least squares sense. The default performance function is the mean square error (MSE), which is defined as

$$\mathrm{MSE} = \frac{1}{N}\sum_{i=1}^{N}(d_i - o_i)^2 = \frac{1}{N}\|\boldsymbol{d} - \boldsymbol{o}\|^2 \tag{4.1}$$

where ||·|| stands for the L^2 vector norm. For a fixed number of training samples, we only need to minimize the function

$$\eta(\boldsymbol{w},\boldsymbol{b},\boldsymbol{\beta}) = \|\boldsymbol{d} - \boldsymbol{o}\|^2 = \|\boldsymbol{d} - \mathbf{H}(\boldsymbol{x},\boldsymbol{w},\boldsymbol{b})\boldsymbol{\beta}\|^2 \tag{4.2}$$

with respect to the parameters $\boldsymbol{w}$, $\boldsymbol{b}$ and $\boldsymbol{\beta}$.

The conventional training process of SLFN involves tuning the parameters $\boldsymbol{w}$, $\boldsymbol{b}$ and $\boldsymbol{\beta}$ using a technique called back propagation [31]. However, it is found that the number of independent parameters can be reduced by making use of the learning algorithm of ELM. It is recognized that the output weights $\boldsymbol{\beta}$ in ELM is obtained by a least squares method. By replacing $\boldsymbol{\beta}$ with the special solution $\boldsymbol{\beta}^* = \mathbf{H}^\dagger \boldsymbol{d}$, the minimization problem takes the form

$$\eta(\boldsymbol{w},\boldsymbol{b}) = \left\|\boldsymbol{d} - \mathbf{H}\mathbf{H}^\dagger\boldsymbol{d}\right\|^2 = \left\|\left(\mathbf{I} - \mathbf{H}\mathbf{H}^\dagger\right)\boldsymbol{d}\right\|^2 \tag{4.3}$$

where $\mathbf{I}$ is the identity matrix. Equation (4.3) indicates that the output weights $\boldsymbol{\beta}$ can be eliminated from the minimization. The original minimization problem is reduced to a smaller one, involving only the input weights $\boldsymbol{w}$ and hidden biases $\boldsymbol{b}$.

Many standard numerical optimization methods can be used to resolve the reduced minimization problem. In this chapter, the Levenberg-Marquardt (LM) algorithm is chosen, which has shown excellent performance for neural network training. The LM inherits the speed advantage of Gauss-Newton method and the stability advantage of steepest descent method, resulting in a very efficient and robust learning process [145].

The LM algorithm is constructed to approach the second-order training speed without computing the Hessian matrix. In fact, the Hessian matrix in LM is estimated using the Jacobian matrix, which is much less complex and can be

easily obtained by the back propagation technique. Following this scheme, the LM algorithm has the update equation

$$\boldsymbol{\varpi}_{k+1} = \boldsymbol{\varpi}_k - [\boldsymbol{\Phi}^{\mathrm{T}}\boldsymbol{\Phi} + \mu\mathbf{I}]^{-1}\boldsymbol{\Phi}^{\mathrm{T}}\boldsymbol{\varepsilon} \tag{4.4}$$

where $\boldsymbol{\varpi}$ contains the input weights and hidden biases, $\boldsymbol{\Phi}$ is the Jacobian matrix that consists of first derivatives of the network errors with respect to the input weights and hidden biases, μ is the damping factor and $\boldsymbol{\varepsilon}$ is the vector of errors. More details about the LM algorithm can be found in [145].

The hybrid learning algorithm can be named as ELM-LM, which has many merits in neural network training. Firstly, it is obvious that ELM is just the first iterative step of ELM-LM. Therefore, ELM-LM is able to obtain better learning accuracy than ELM. Secondly, unlike the conventional SLFNs that all the parameters (i.e. $\boldsymbol{w}$, $\boldsymbol{b}$ and $\boldsymbol{\beta}$) are modified by the optimization method (e.g. LM), the proposed hybrid algorithm only applies LM to adjust $\boldsymbol{w}$ and $\boldsymbol{b}$. It is noted that the reduced minimization problem has fewer independent parameters which need to be estimated. Therefore, the time taken to reach a minimum of the reduced problem will be less than that for the original problem. It should be noted that the time of least squares method can be neglected in comparison with that of LM. Furthermore, since $\boldsymbol{\beta}$ is obtained by a least squares method, the training error is always at a global minimum with respect to $\boldsymbol{\beta}$. The robustness of the learning procedure is improved [124].

The ELM-LM process can be described as follows:

1) Initialize the input weights $\boldsymbol{w}$ and hidden biases $\boldsymbol{b}$ with a set of random numbers.

2) Compute the matrix **H** and the output weights $\boldsymbol{\beta}$.
3) Start the learning process with the modified performance function, as shown in (4.3).
4) Update $\boldsymbol{w}$ and $\boldsymbol{b}$ using the LM algorithm and determine $\boldsymbol{\beta}$ using the least squares method.
5) The early stopping criterion is used to terminate the training process.

It is noted that ELM-LM and ELM-MABC (in Chapter 3) have similarities. Both of them are iterative learning algorithms for feedforward neural networks. In each iterative step, the output weights of hidden layer are calculated using the same least squares method from ELM. The difference is the method used for the update of input weights and hidden biases. They employ LM and MABC, respectively.

4.2.2 Wavelet-Based Ensemble Scheme

The wavelet transform is very effective in uncovering the characteristics of load data [110]. As mentioned in Chapter 3, two important parameters must be supplied: mother wavelet and number of decomposition levels. So far, there is no single fixed criterion concerning the selection of these parameters. Many papers have tested various wavelet specifications and selected the most suitable setting according to the results. However, this kind of selection process brings at least three shortcomings. Firstly, the testing of various wavelet specifications would take too much time. Only the best wavelet setting is selected and the others are abandoned, which is undesirable. Secondly, the fixed specification may not always represent the load series adequately. Since the load series varies all the time, the optimal setting obtained from one load segment may not be

good for another segment. Thirdly, fixed wavelet parameters cannot always obtain the best forecasting result in all horizons, which will be demonstrated in Case 3.

In this chapter, a wavelet-based ensemble scheme is proposed to overcome the above problems. An ensemble of predictors is generated using the wavelet transform, in which each predictor is featured with a different combination of mother wavelet and number of decomposition levels. In STLF, the wavelet family of Daubechies (db) has been widely used to treat the load data [103, 107]. In the testing, it is found that Coiflets (coif) can also give a good representation, although they rarely appear in the literature. Moreover, the mother wavelet order typically ranges from 2 to 5 and the number of decomposition levels is less than 4. In this chapter, 8 wavelet functions (db2–db5 and coif2–coif5) are therefore selected and the level for decomposition is from 1 to 3. Overall, the proposed ensemble scheme consists of 24 different combinations of wavelet parameters.

Four mother wavelets (db2, db4, coif2 and coif4) are shown in Figure 4.1. It is observed that a wavelet function only fluctuates in a limited amount of time. Since the wavelet functions are in different shapes, their corresponding sub-components will also have different behaviors. For example, if a load signal is decomposed into three levels using the above four wavelet functions, then the resulting approximation and detail components are shown in Figures 4.2-4.5.

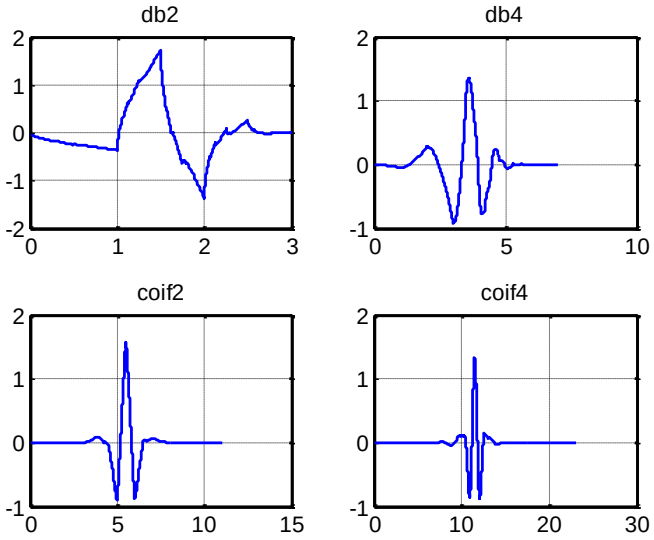


Figure 4.1 Four mother wavelet functions.

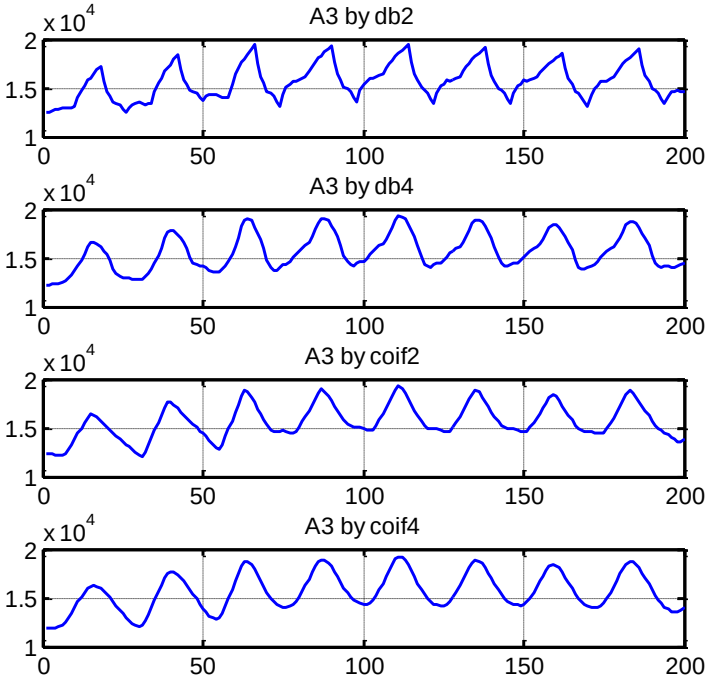


Figure 4.2 Approximation components obtained by db2, db4, coif2 and coif4.

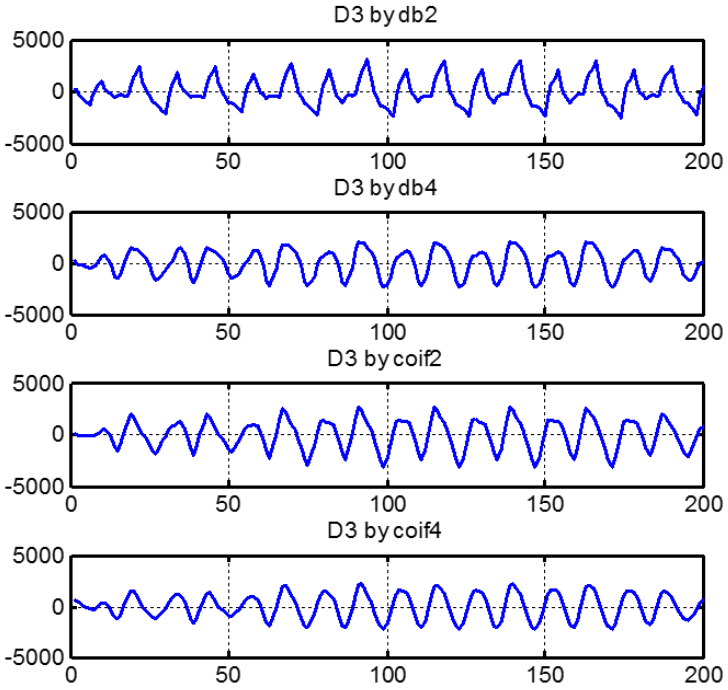


Figure 4.3 Detail components D_3 obtained by db2, db4, coif2 and coif4.

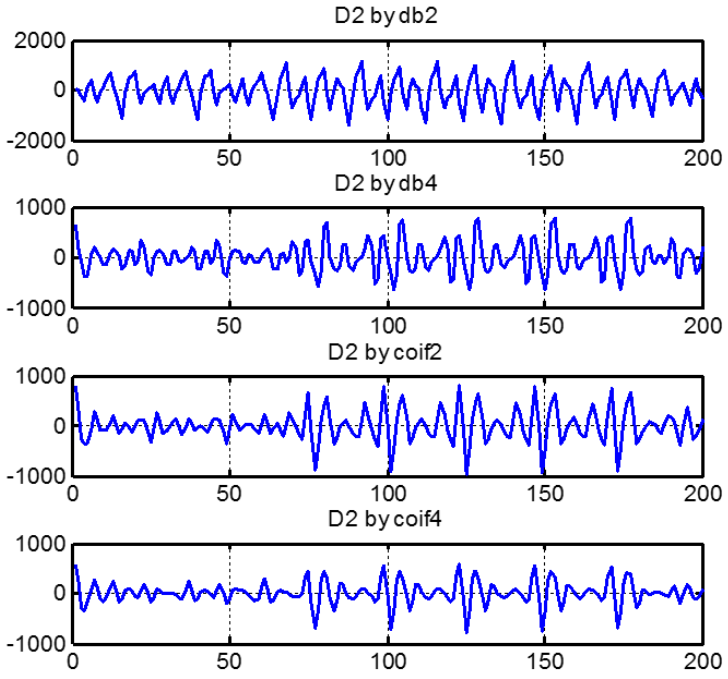


Figure 4.4 Detail components D_2 obtained by db2, db4, coif2 and coif4.

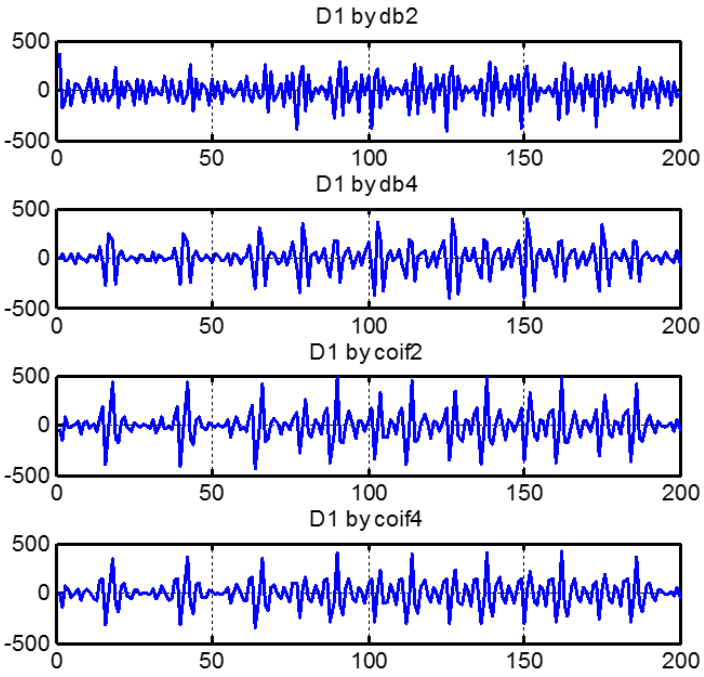


Figure 4.5 Detail components D_1 obtained by db2, db4, coif2 and coif4.

The wavelet-based ensemble scheme will generate different input variables for individual forecasters, which can encourage the diversity of the ensemble. One individual forecast is raised based on the variables that are different from those for another forecast. Hence, every forecast in the ensemble would have some independent information. The ensemble strategy can take advantage of the useful complementary information to improve the generalization capability. Moreover, the ensemble strategy will avoid the selection process for optimal wavelet parameters and suppress the biases induced by fixed parameters.

4.2.3 Partial Least Squares Regression

Simple averaging is commonly used to aggregate the individual forecasts in STLF [95, 142], which assigns identical weight factors to individual outputs. Using simple averaging, the ensemble output is given by

$$z_e = \frac{1}{\gamma}\sum_{i=1}^{\gamma} z_i \tag{4.5}$$

where z_e is the ensemble output, z_i is the individual output and γ is the number of individual outputs.

However, in practice, some individual predictors in the ensemble have better accuracy than the others. The relative accuracy of the individual forecasts should be considered. In order to identify the unique contribution of each individual forecast, we can assign different weight factors to them and produce the ensemble output by

$$z_e = \frac{1}{\gamma}\sum_{i=1}^{\gamma} \alpha_i z_i \tag{4.6}$$

where α is the weight factor. In this chapter, partial least squares regression (PLSR) is used to calculate the optimal combining weight factors.

The idea of PLSR is to find a linear regression model by projecting the original variables to a new space. Supposing **M** contains the individual outputs and **R** denotes the ensemble output, PLSR is going to predict **R** from **M** by latent variables. In order to make the regression efficient, **M** and **R** are tailored by mean-centering and scaling, which are denoted by $\mathbf{M_c}$ and $\mathbf{R_c}$, respectively. For each column of **M**, mean-centering involves subtracting its column average from the column data. The scaling of a matrix makes each column has standard deviation 1 [146].

Using the outer relations [147], PLSR decomposes $\mathbf{M_c}$ and $\mathbf{R_c}$ by

$$\begin{aligned}\mathbf{M_c} &= \mathbf{TP}^{\mathrm{T}} + \mathbf{E} = \sum_{j=1}^{J} \boldsymbol{t}_j \boldsymbol{p}_j^{\mathrm{T}} + \mathbf{E} \\ \mathbf{R_c} &= \mathbf{UQ}^{\mathrm{T}} + \mathbf{F} = \sum_{j=1}^{J} \boldsymbol{u}_j \boldsymbol{q}_j^{\mathrm{T}} + \mathbf{F}\end{aligned} \tag{4.7}$$

where **T** and **U** are the score matrices, **P** and **Q** are the loading matrices, **E** and **F** are the residual matrices and J is the number of components. The meanings of score and loading matrices are too complex to explain here, but can be found in [146, 147]. PLSR intends to search for the latent vectors $\boldsymbol{t}_j$ and $\boldsymbol{u}_j$ to maximize the covariance between $\mathbf{M_c}$ and $\mathbf{R_c}$ with the condition that the residuals **E** and **F** are reduced [147].

It is seen that the matrices $\mathbf{M_c}$ and $\mathbf{R_c}$ have been replaced by the new ones **T** and **U** in (4.7), which have smaller sizes, better properties (orthogonality) and also span the spaces of $\mathbf{M_c}$ and $\mathbf{R_c}$. PLSR builds an inner relation between **T** and **U** by

$$\mathbf{U} = \mathbf{TB} + \mathbf{U_E} \tag{4.8}$$

where **B** is the matrix of regression coefficients and $\mathbf{U_E}$ refers to an error term similar to **E** and **F**.

If the error terms are minimized, we can estimate $\mathbf{R_c}$ by

$$\hat{\mathbf{R}}_{\mathbf{c}} = \mathbf{UQ}^{\mathrm{T}} = \mathbf{TBQ}^{\mathrm{T}}. \tag{4.9}$$

The matrix of individual outputs **M** is finally related to the matrix of ensemble output **R** via a series of intermediate variables. The whole process is shown in Figure 4.6. In PLSR, the solutions to the above equations can be approached by the SIMPLS method [146]. In our experiments, we can directly use the built-in

function *plsregress* in Matlab to carry out PLSR [148]. In order to clarify the procedure of PLSR, an example of deriving ensemble output is presented in the Appendix.

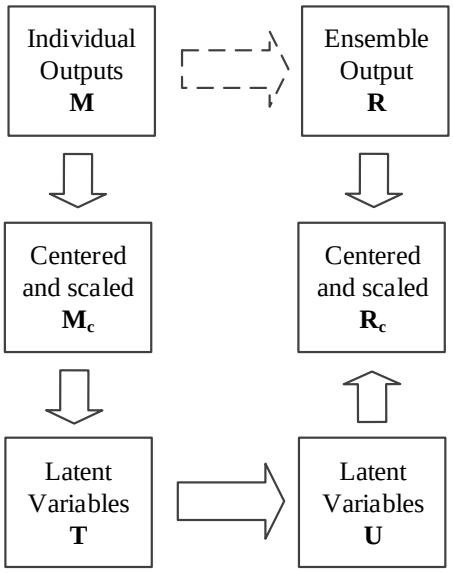


Figure 4.6 Overview of PLSR.

It should be noted that the number of components J in (4.7) is an important parameter in the regression. Normally, it is not appropriate to use as many components as possible. The main reasons for this are that the observed data are never noise-free and some of the smaller components only describe the noise, which may bring the problems of collinearity and overtraining. In this chapter, cross validation is used to select the components. The prediction residual sum of squares (PRESS) is calculated each time a new latent component is added to the PLSR model.

In order to determine the combining weight factors, the regression is not performed on the original variables but on the latent variables, which have

better properties. Hence, PLSR can handle the high correlation between the individual forecasts and generate reliable estimates for the combining weights. Moreover, PLSR allows selecting a subset of components with the best estimate of generalization performance, which can get rid of overfitting (including too many components) the individual forecasts [147].

4.2.4 Conditional Mutual Information Based Feature Selection

In information theory, the entropy $H(X)$ is a measure of the uncertainty of the random variable X. The entropy does not depend on the actual values taken by X, but only on the probability distribution of X [149]. The conditional entropy $H(X|Y)$ represents the remaining uncertainty of X after Y is known. The mutual information (MI) between X and Y is given by [149]

$$\mathrm{MI}(X;Y) = H(X) - H(X \mid Y) \tag{4.10}$$

which indicates that MI(X;Y) quantifies how much uncertainty of X is reduced if Y is known. The MI gives an estimate of the quantity of dependence of two variables. High MI suggests that the two variables are strongly correlated and vice versa.

The conditional mutual information (CMI) between X and Y given Z is defined by

$$\mathrm{MI}(X;Y \mid Z) = H(X \mid Z) - H(X \mid Y,Z). \tag{4.11}$$

The CMI denotes the amount of information shared between X and Y when Z is known. If Y and Z bring the same information about X, $H(X|Z)$ and $H(X|Y,Z)$ are equal and MI(X;$Y|Z$) is zero. Otherwise, if Y holds information about X that Z does not have, the difference on the right is large and so is MI(X;$Y|Z$).

Let C be the dependent variable and $V=\{V_1, V_2,\ldots, V_{NC}\}$ be the candidate set with NC features. The goal of feature selection is to choose a subset $S \subset V$ with NS features such that the conditional entropy $H(C|S)$ is minimized. In other words, knowing the subset S can maximally reduce the uncertainty of C. But in practice, the minimization of $H(C|S)$ is computationally intractable as the total number of subsets for evaluation is extremely large [150]. Hence, approximated solutions are developed for feature selection in applications.

Fleuret proposed an approximated feature selection method based on CMI, which can guarantee that the selected features are both individually informative and mutually weakly dependent [151]. In this technique, the first feature to be selected is the one that holds the highest MI value with C. The highest MI value means that this feature is the most correlated to C. From now on, a feature V_j is considered good if it carries information about C and this information has not been caught by the features already selected. The selection criterion depends on how much information the new feature V_j can add with respect to the selected features. The next feature V_j to be selected is the one that makes $\mathrm{MI}(C;V_j|V_i)$ large for every selected feature V_i. Following this rule, the feature is chosen one by one until the subset S is full of NS features.

The above feature selection approach has two problems that need to be addressed. Firstly, it is possible to pick bad features if we only use the measure $\mathrm{MI}(C;V_j|V_i)$. Based on the concept of trivariate mutual information, $\mathrm{MI}(C;V_j|V_i)$ can be expressed by

$$\mathrm{MI}(C;V_j \mid V_i) = \mathrm{MI}(C;V_j) - \mathrm{MI}(C;V_j;V_i) \tag{4.12}$$

where MI(C;V_j;V_i) measures the amount of information shared by C, V_j and V_i. It is observed that the large value of MI(C;V_j|V_i) can be obtained by two small values MI(C;V_j) and MI(C;V_j|V_i). Note that the small value MI(C;V_j) implies that the new feature V_j is weakly correlated to C, which is harmful to the model. To tackle this problem, a preliminary selection process for relevant features is proposed. In this step, the MI values are calculated for every candidate feature and ranked from high to low. Then the features in the top T places are retained and the others are removed from the candidate set. The new set is used for further selection by the Fleuret's approach. Note that T is a threshold parameter for filtering out the irrelevant and weakly relevant features. In the literature, the number of features used in STLF is at most twenty or thirty. Thus, T is set to be fifty in this chapter, which is deemed enough to involve all relevant features.

Secondly, it is clear that the estimate of MI values is crucial for the efficacy of feature selection. In [151], Fleuret's approach was employed to select binary features for classifiers. However, the input and output variables in STLF are continuous, making the estimation more difficult. In this chapter, the estimation method in [152] is employed, which can obtain the MI values accurately and efficiently.

In this chapter, the candidate input features contain the past load data, the past and forecasted temperature readings and the calendar variables. Therefore, a candidate set of input features, denoted by $V(t)$, is given by

$$\begin{aligned} V(t) = \{&\text{Load}(t-1), \text{Load}(t-2), ..., \text{Load}(t-200), \\ &\text{Temp}(t), \text{Temp}(t-1), ..., \text{Temp}(t-200), \\ &\text{Dayofweek}(t), \text{Weekend}(t)\} \end{aligned} \tag{4.13}$$

where t is the time index. As hourly load forecasting is studied in this chapter, t is on an hourly basis. In (4.13), Load(t-1), …, Load(t-200) are the historical load values, because the load data show short-run trend and daily and weekly periodicities. Temp(t),…, Temp(t-200) are the current and historical temperature values, because temperature is a very crucial weather factor in load forecasting. The order of historical values is set to be 200, which is sufficient to contain all useful features. Dayofweek(t) refers to the day of the week, which is marked by the numbers from 1 to 7. For example, 1 is used for Monday and 7 is used for Sunday. Weekend(t) uses binary numbers to label weekend (1) and weekday (0). It is noted that all the public holidays are regarded as weekends and marked by 1. Therefore, there are 403 input features in the candidate set $V(t)$.

Assuming that the subset S to be determined consists of NS features, the proposed CMIFS method can be summarized as follows:

1) Find the dependent variable C (i.e., load vector) and its corresponding candidate set V of NC (=403) features.
2) For each feature $V_j \in V$, compute MI(C;V_j) and rank.
3) Move the top T (=50) features to a new set V_{new}.
4) Initialize the selected subset $S = \{\emptyset\}$.
5) Find the feature V_k in V_{new} that maximizes MI(C;V_k). Set $V_{new} \leftarrow V_{new} \setminus \{V_k\}$ and $S \leftarrow S \cup \{V_k\}$.
6) For all pairs of features (V_i, V_j) with $V_i \in S$ and $V_j \in V_{new}$, compute MI(C;V_j|V_i).
7) Select the next feature V_k, which is defined by

$$V_k = \arg \max_{V_j \in V_{new}} \min_{V_i \in S} \left\{ \mathrm{MI}(C; V_j \mid V_i) \right\}. \quad (4.14)$$

8) Set $V_{new} \leftarrow V_{new} \setminus \{V_k\}$ and $S \leftarrow S \cup \{V_k\}$.

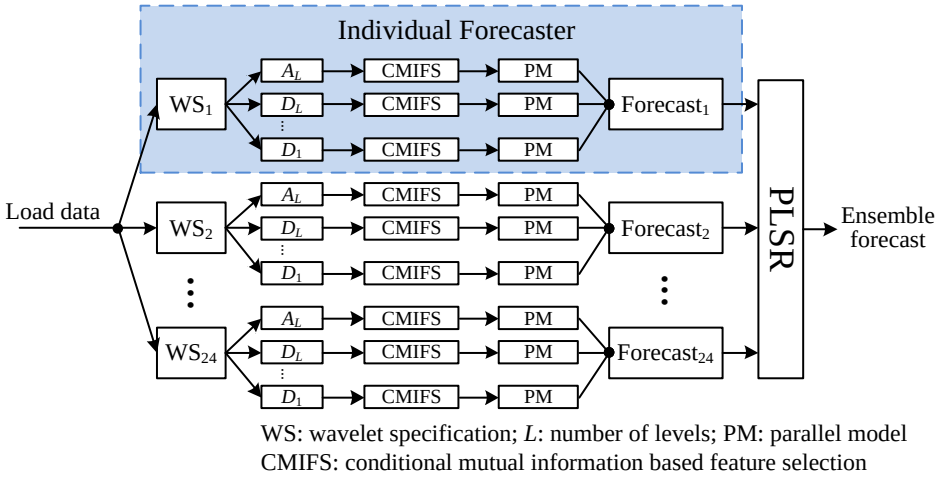
9) Repeat steps 6) to 8) until S is full of NS features.

10) Output the subset S containing the selected features.

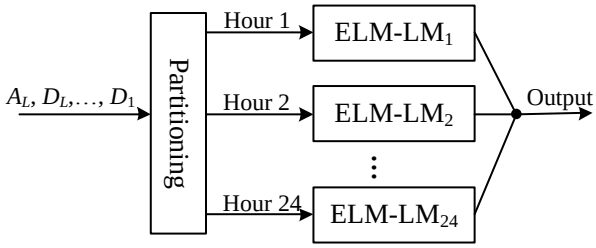
It is seen that the proposed CMIFS method can be divided into two phases. In the first phase, the irrelevant or weakly relevant features are removed based on the predefined threshold T. The retained features are also ordered from high MI to low MI. In this phase, the number of features is greatly reduced from 403 to 50. In the second phase, a new feature is selected considering not only the relevance to the dependent variable C but also the redundancy to the existing features. The time taken in the second phase will be significantly shorten due to the reduction in number of features. In order to clarify the procedure of CMIFS, an example of feature selection is presented in the Appendix.

4.2.5 Proposed Model

The proposed model is shown in Figure 4.7, which includes wavelet-based ensemble scheme, variable selection, hybrid neural networks, parallel forecast model and partial least squares regression.



(a) Ensemble model



(b) Parallel model

Figure 4.7 Overview of the proposed ensemble method.

In the individual forecaster, wavelet transform is used to decompose the load into one approximation component associated with the low frequency and a group of detail components associated with the high frequency. In order to establish the ensemble model, 24 sets of wavelet specifications (WS_1, WS_2,…, WS_{24}) are employed for the load decomposition.

For each sub-component (A_L, D_L, …, D_1) obtained from the wavelet decomposition, the proposed CMIFS technique is used to decide a small set of input variables by removing the weakly relevant and redundant ones from the candidate set defined in (4.13). Then a parallel model (Figure 4.7b) including

24 feedforward neural networks is allocated to estimate the hourly loads of the next day. It is obvious that finding a small and precise network for one hour is much easier than finding a network for one day. Moreover, since the networks are relatively small, they are not likely to be overtrained in the learning process. To implement the network, the sigmoid and linear transfer functions are used in the hidden and output layers, respectively. To boost the learning performance, a hybrid algorithm named ELM-LM is proposed to train the networks. To avoid the overtraining problem, the early stopping method is employed to control the training process [31]. It is noted that the number of selected variables and the number of hidden nodes are decided through extensive tests.

It is evident that the number of individual forecasters is the same as the number of combinations of wavelet parameters. The 24 individual forecasts are linearly combined through PLSR, which can produce an accurate and reliable ensemble forecast. It is noted that the number of components used in PLSR is determined by the cross validation method.

4.3 Results

In this section, the proposed ensemble model is tested using actual load and temperature data. In order to confirm the effectiveness and superiority, the proposed method is compared with other state-of-the-art methods based on public datasets. It should be noted that all data are publicly available which allows other researchers to implement comparison and reproduction. In this chapter, in addition to MAPE and MAE, root mean square error (RMSE) is also employed to evaluate the forecasting performance. The RMSE is defined by

$$\text{RMSE} = \sqrt{\frac{1}{m}\sum_{i=1}^{m}\left(a_i - \hat{a}_i\right)^2} \tag{4.15}$$

where m is the number of load data points, a_i is the actual load value and $\hat{a}_i$ is the forecasted load value.

a) Case 1

In this case, the hourly data from ISO New England were used. Two typical months have been selected to forecast. The first one refers to January 2010, which is a winter month. The second one refers to July 2010, which is a summer month. For the winter month, the training set is from January 1, 2009 to December 31, 2009. For the summer month, the training set is from July 1, 2009 to June 30, 2010. The validation set is the last month of training set, which is used to fine-tune the model parameters.

The hourly load data from January 1, 2009 to July 31, 2010 are shown in Figure 4.8. It is seen that the load varies greatly between 9040 MW and 27102 MW. The mean values of two testing months (15550 MW for winter and 17989 MW for summer) are both larger than 14664 MW. The reason may be that the heating and cooling loads are high in these two months.

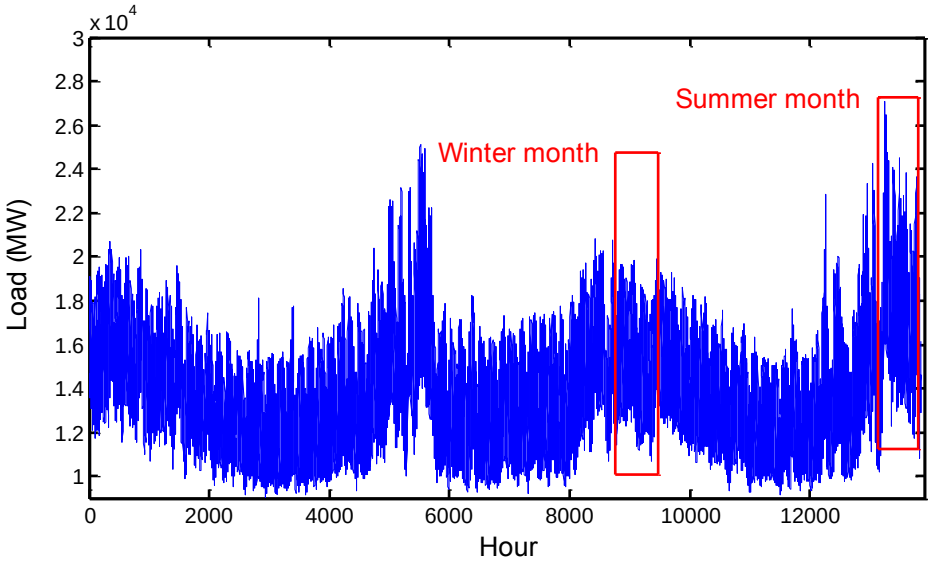


Figure 4.8 Hourly load data from ISO New England.

Four forecast models are chosen for the purpose of comparison:

1) **M1**: Wavelet transform is used to decompose the load data and each sub-component is modeled by an ELM.

2) **M2**: All the sub-components in M1 are forecasted using the parallel model, which is made up of 24 ELMs.

3) **M3**: The proposed ensemble scheme is employed to generate individual forecasters, which are then combined to produce an ensemble forecast by simple averaging.

4) **M4**: PLSR is used to combine the individuals in M3.

It should be mentioned that the input variables in the models M1-M4 are selected by the correlation analysis. The 1-hour and 24-hour ahead forecasting results are shown in Tables 4.1 and 4.2, respectively.

Table 4.1 1-hour ahead forecasting results of M1-M4 and the proposed method

	Winter month			Summer month		
	MAPE (%)	MAE (MW)	RMSE (MW)	MAPE (%)	MAE (MW)	RMSE (MW)
M1	0.4637	71	103	0.7566	139	187
M2	0.4549	70	101	0.6909	125	168
M3	0.3482	55	74	0.5586	109	144
M4	0.2736	43	59	0.5227	103	134
Proposed	0.1331	21	34	0.2229	42	63

Table 4.2 24-hour ahead forecasting results of M1-M4 and the proposed method

	Winter month			Summer month		
	MAPE (%)	MAE (MW)	RMSE (MW)	MAPE (%)	MAE (MW)	RMSE (MW)
M1	2.1884	332	474	4.2495	775	1058
M2	1.8662	289	402	3.6552	678	957
M3	1.5035	231	322	2.9883	538	749
M4	1.4279	220	303	2.8158	504	708
Proposed	1.1075	170	237	2.3703	433	582

The findings can be summarized as follows:

1) As shown in Tables 4.1 and 4.2, the forecasting accuracy of winter month is much better than that of summer month. The reason may be that the summer load data have more irregular fluctuations, which can be seen in Figure 4.8.

2) The performance is improved if the sub-components from the wavelet transform are forecasted using the parallel model (M2) instead of the single ELM model (M1).

3) The forecast results show that the proposed ensemble strategy is an effective way to enhance the performance. For example, M3 has obtained an increase of 23.5% (in MAPE) over M2 in

the 1-hour case of winter.

4) PLSR has shown better accuracy as compared to the simple averaging method. Taking the summer month as an example, the forecast accuracy of the 24-hour case has been improved by 5.8% (in MAPE) from M3 to M4. This implies that PLSR is able to produce more accurate ensemble forecast than the simple averaging method.

5) Compared to M4, the proposed method can greatly enhance the prediction accuracy. The improvements can be attributed to the integration of CMIFS and ELM-LM.

6) The proposed method has surpassed other methods in all cases, which proves its feasibility and effectiveness in STLF. Taking the 24-hour case of winter as an example, the enhancements of the proposed method over M1-M4 are 49.4%, 40.7%, 26.3% and 22.4%, respectively.

The forecast results of the proposed method are shown in Figures 4.9 and 4.10.

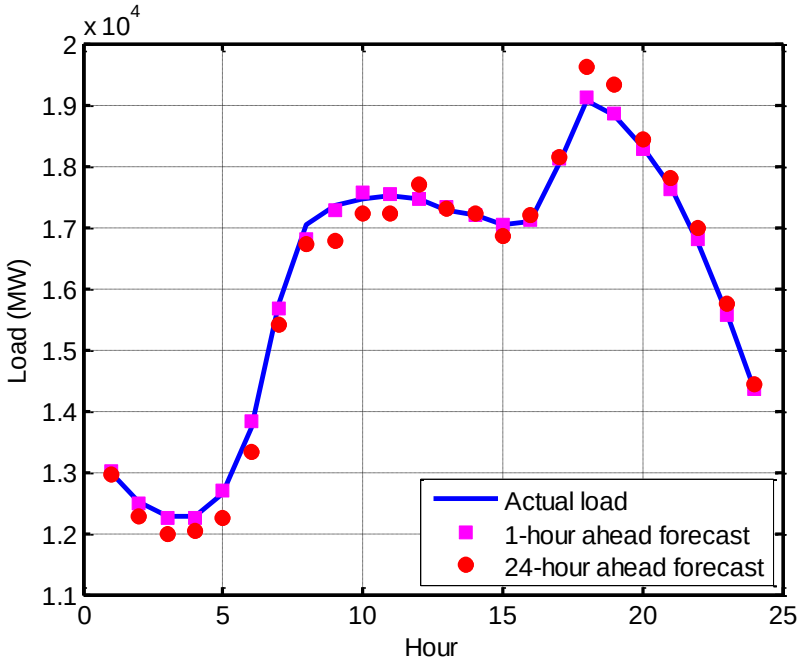


Figure 4.9 Forecast results of a winter day.

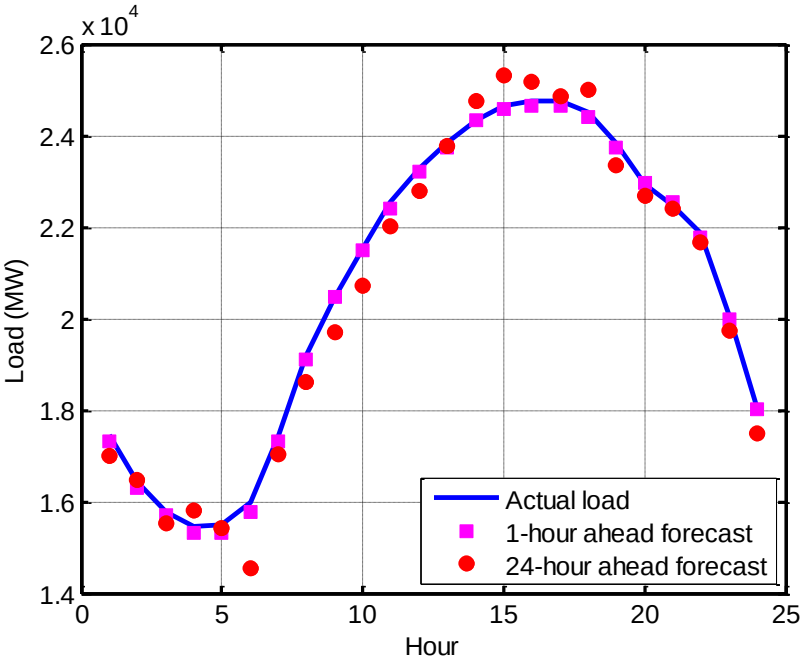


Figure 4.10 Forecast results of a summer day.

b) Case 2

This case compares the proposed CMIFS approach to other feature selection methods using the data in Case 1. For each sub-component (A_L, D_L,…, D_1) obtained from the wavelet decomposition, the same number of input variables is retained by the feature selection methods. The results for 1-hour ahead forecasting are shown in Table 4.3. The benchmark feature selection methods include correlation analysis (CA), mutual information (MI) and RReliefF [153]. Only the input variables with top *scores* (e.g. correlation coefficient for CA) are selected by the three benchmark methods. The proposed CMIFS method is different in that it also manipulates the feature redundancy by comparing the new feature with the ones already selected.

Table 4.3 1-hour ahead forecasting results (in MAPE) of four feature selection methods

	Winter	Summer
CA	0.2012	0.3101
MI	0.2295	0.3709
RReliefF	0.2498	0.3769
CMIFS	0.1331	0.2229

As shown in Table 4.3, the best forecasting performance over the testing period is yielded by the proposed CMIFS method. For example, CMIFS improves the winter results by 33.8%, 42.0% and 46.7% with respect to the previous three methods, respectively. The significant improvements over CA, MI and RReliefF have shown that relevance analysis alone is not sufficient for feature selection.

c) Case 3

The forecasting efficacy of the proposed wavelet-based ensemble scheme is further investigated in this case. The individual forecast utilizes a specific set of wavelet parameters while the ensemble forecast aggregates the individual forecasts by PLSR. The forecasting results of winter month at four look-ahead times are shown in Table 4.4. In this table, "F16" refers to the ensemble forecast derived from the first sixteen individuals.

Table 4.4 Individual and ensemble forecasting results (in MAPE) at four different look-ahead times

Level	Wavelet	1-hour	6-hour	12-hour	24-hour
1	coif2	0.1776	2.6164	3.1785	1.4677
	coif3	0.1993	2.9190	2.2047	1.8395
	coif4	0.2317	2.1185	2.2659	1.0227
	coif5	0.2660	2.3379	1.8841	1.5328
	db2	0.3822	1.8634	2.0865	1.2043
	db3	0.2687	2.2408	2.0483	1.3847
	db4	0.2698	1.5494	2.7330	1.6375
	db5	0.1768	1.7039	3.3935	1.3612
2	coif2	0.3190	2.1157	1.9689	1.3871
	coif3	0.4130	1.9372	2.5723	1.6032
	coif4	0.3494	2.2139	2.6348	2.4805
	coif5	0.2816	1.9577	2.8441	1.8832
	db2	0.2964	1.9433	2.4877	1.3821
	db3	0.3502	3.1411	3.4737	1.0562
	db4	0.4050	1.8288	4.0573	1.3026
	db5	0.2348	1.7291	2.2124	1.3574
3	coif2	0.6357	0.8253	1.7945	1.5199
	coif3	0.8599	2.1053	3.4794	1.7919
	coif4	0.6112	0.9298	1.4452	1.2440
	coif5	0.8139	1.1555	1.5473	1.3095
	db2	0.6878	2.9649	3.1023	1.2311
	db3	0.9453	1.3241	2.5686	3.2367
	db4	0.5305	0.9382	2.1026	1.4398
	db5	1.1783	1.5295	2.2106	1.4161
F16		0.1181	0.8657	1.5752	0.8850
Proposed		0.0790	0.6120	1.0423	0.7530

It is obvious that the forecasting performance varies greatly with the wavelet parameters. A fixed set of wavelet parameters cannot yield

the best results in all horizons. Moreover, the two ensemble forecasts have provided better results than any of the individual forecasts, which confirms the effectiveness of the proposed ensemble scheme. Besides, the proposed method can deliver better forecasting result than F16 even with some poor individual forecasts (i.e., large MAPEs). This indicates that the poor individuals still contain some information that is useful to the ensemble forecast.

d) Case 4

This case compares the proposed method with the standard neural network (SNN) model [129], the similar day-based wavelet neural network (SIWNN) model [129] and the random forest model [154]. The SNN method was implemented by using a single neural network, which used weekday index, weather and historical load values as the inputs. The SIWNN method selected similar day load as the input data and applied wavelet neural networks to capture the load features at low and high frequencies. The random forest is an ensemble method which involves a multitude of regression trees. The number of regression trees is set to be 24, which is equal to the number of individual forecasters in the proposed method.

The comparison is implemented on the ISO New England data. The training period is from March 2003 to December 2005. The hourly loads from January 1, 2006 to December 31, 2006 are forecasted. The 24-hour ahead forecast results are given in Figure 4.11, where the values of SNN and SIWNN are extracted from [129]. The forecast

results of the random forest method were obtained through simulation. It is clear that the proposed method provides better results than other methods in all months. On an average, the proposed method is 33.6%, 21.0% and 36.6% better than the SNN, SIWNN and random forest methods, respectively.

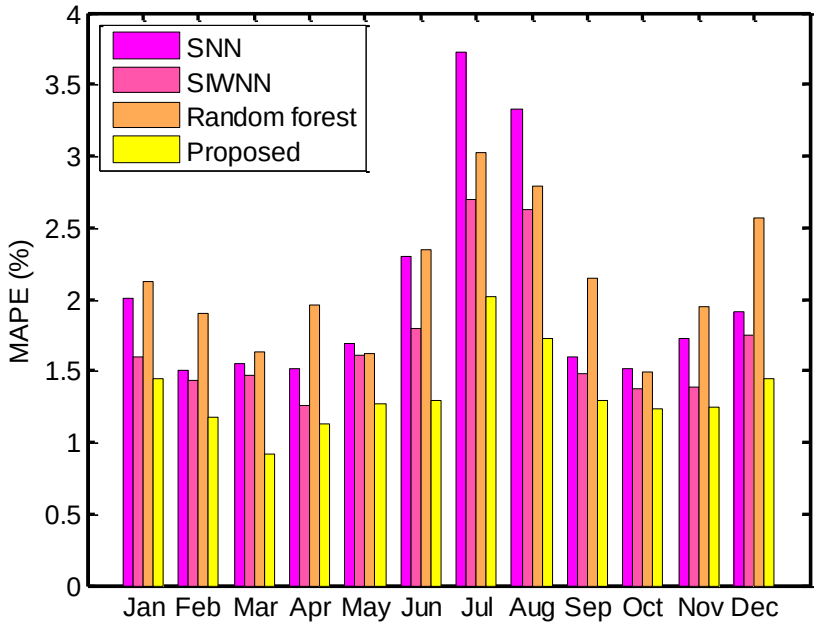


Figure 4.11 MAPE results of SNN, SIWNN, random forest and proposed method.

e) Case 5

The proposed method is compared to the ISO-NE method [127], the wavelet neural networks (WNN) method [128], the multiple linear regression (MLR) model [155] and the radial basis function neural network (RBFNN) method [135] based on the ISO New England data. The testing period is from July 1, 2008 to July 31, 2008. Only the 1-hour ahead load forecasting is studied. The forecast results are shown in

Table 4.5. The results of MLR and RBFNN were obtained through simulation while the results of ISO-NE and WNN were directly taken from [128]. For the given testing month, the proposed method has provided great improvements as compared to the other four methods. For example, the improvements in MAE of the proposed method with respect to the previous methods are 73.9%, 57.1%, 67.0% and 60.9%, respectively.

Table 4.5 Forecast results of ISO-NE, WNN, MLR, RBFNN and the proposed method

	MAPE (%)	MAE (MW)
ISO-NE	0.81	138
WNN	0.49	84
MLR	0.64	109
RBFNN	0.54	92
Proposed	0.21	36

f) Case 6

The performance of the proposed method was examined based on the actual data of an electric utility in North America. The experiment was conducted using the load and temperature data from January 1, 1988 to October 12, 1992. The two-year load values prior to October 12, 1992 were forecasted and the remaining data were used for building the model.

The proposed model was compared to other state-of-the-art models in [107, 110, 130, 131]. Both the 1-hour and 24-hour ahead forecasts were studied. In order to probe the effect of temperature forecasting errors on STLF, the proposed method was examined using the noisy temperatures. The Gaussian noise of zero mean and standard deviation

of 0.6 ºC was added to the actual temperature data [107].

The 1-hour and 24-hour ahead forecast results are shown in Table 4.6. The improvement of the proposed method compared to the best available method is given in the last row. It is seen that the proposed method has provided better results than other methods in all cases. It should be pointed out that the results of [107] and [110] did not include public holidays and weekends. Moreover, the proposed method can also achieve encouraging results in case of noisy temperatures. The 24-hour forecast results of the proposed method are presented in Figure 4.12.

Table 4.6 MAPE (%) values for the models in Case 6

	Actual temperature		Noisy temperature	
	1-hour	24-hour	1-hour	24-hour
Model 1 [107]	2.10	3.58	2.15	4.46
Model 2 [107]	1.10	3.41	1.11	3.64
Model 3 [107]	1.12	3.16	1.14	3.38
Model 4 [107]	1.99	2.64	2.04	2.82
WT-NN-EA [110]	0.99	2.04	n.a.	n.a.
ESN [130]	1.14	2.37	1.21	2.53
SSA-SVR [131]	0.72	1.99	0.73	2.03
Proposed	0.38	1.67	0.40	1.73
Improvement	47.2%	16.1%	45.2%	14.8%

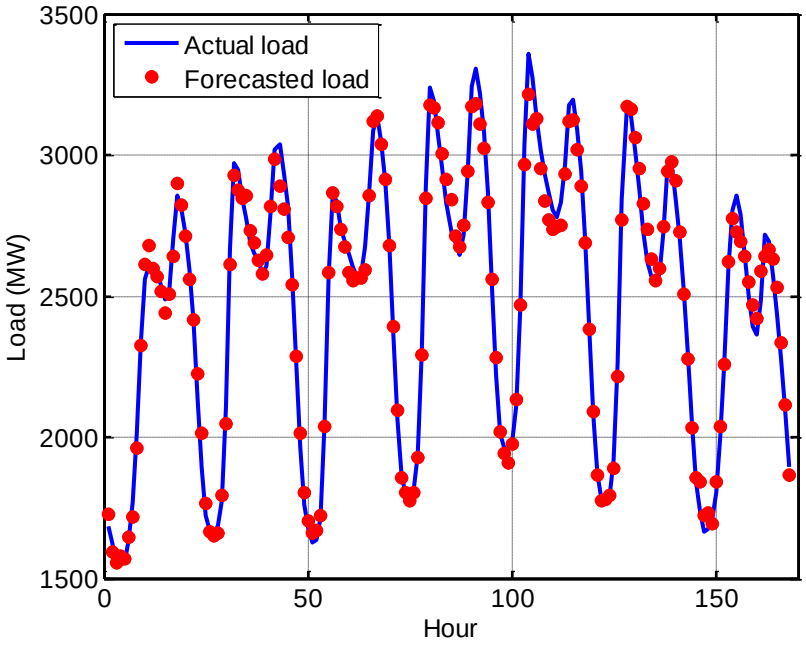


Figure 4.12 Forecast results of a week using the actual temperature.

g) Case 7

In this case, a comparison between the proposed method and the abductive network model [156] was performed on the North American dataset. Furthermore, the random forest model used in Case 4, and the MLR and RBFNN models used in Case 5 were also considered here. The five-year (1985-1989) data were used for model synthesis and the forecast model was tested on the next year (1990). The forecast results of 1-hour and 24-hour cases are given in Table 4.7. It is clear that the proposed method shows significant improvements over other forecast models for both 1-hour and 24-hour cases. The percentage increase of the proposed method with respect to the best available method is given in the last row. Figure 4.13 illustrates the forecast accuracy at different

horizons.

Table 4.7 MAPE (%) values for the models in Case 7

	1-hour	24-hour
Abductive	1.14	2.66
MLR	1.18	2.89
RBFNN	0.91	2.40
Random forest	1.27	3.19
Proposed	0.35	1.84
Improvement	61.5%	23.0%

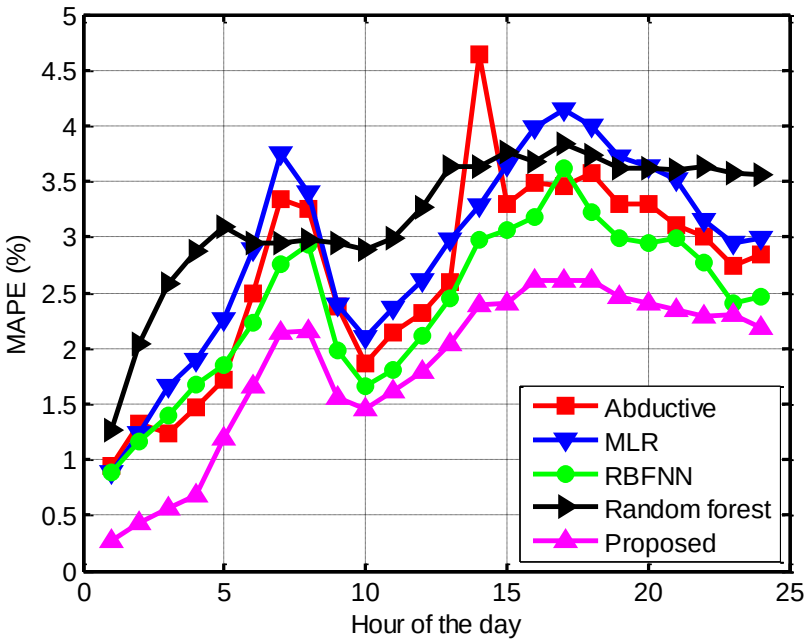


Figure 4.13 Forecast results of the models in Case 7.

4.4 Conclusion

In this chapter, a novel ensemble model for STLF is proposed based on the ELM. In order to increase the forecasting accuracy, several improvements are exploited to support the ELM. Firstly, a novel wavelet-based ensemble strategy is used to build the neural network ensemble model. Secondly, a hybrid learning algorithm called ELM-LM is proposed to improve the learning performance of

SLFNs. Thirdly, a conditional mutual information based feature selection (CMIFS) method is developed to select the most informative input variables for the STLF model. Fourthly, to realize an accurate and reliable ensemble forecast, PLSR is used as a combination method to aggregate the individual forecasts.

The proposed method is able to alleviate many difficulties such as random network weights and biases, wavelet parameter determination, uncertainty and overtraining. The improvements can be attributed to several factors such as the proposed ensemble strategy, the proposed feature selection technique, the early stopping criterion, the proposed hybrid learning algorithm, and the partial least square regression method.

The performance of the proposed method was tested using actual data from two utilities. Both 1-hour and 24-hour ahead load forecasts were considered. The experimental results demonstrate that the proposed model can significantly improve the forecasting accuracy. Compared to other state-of-the-art methods, the proposed model has shown better forecasting performance.

Chapter 5 Wind Power Forecasting Using Neural Network Ensembles with Feature Selection

5.1 Introduction

Wind power is rapidly becoming a popular renewable energy technology. However, the randomness and intermittency of wind present some difficulties to integrate wind power into the electrical systems [11]. An effective solution for such problems is to set up an accurate forecasting system for wind power generation. Various parties can benefit from accurate wind power forecasting results. For example, system operators can make use of the wind power forecast results to maintain the balance between supply and load, and determine the best management strategy for energy storage systems in the grid [59].

Wind power forecasting (WPF) methods can be roughly classified into two groups: physical approach and statistical approach. The physical approach takes topological and meteorological variables as the input data to generate the best possible wind speed predictions. The forecasted wind speeds are then converted into wind power forecasts via the power curves of wind turbines. The statistical

approach tries to establish a relationship between the wind power and a set of variables such as historical wind power data and historical and forecasted wind speed and direction values. The physical approach has advantages in long-term forecasts (6 hours ahead and more) while the statistical approach performs well in short-term [62]. In addition, the hybrid of physical and statistical methods is also an alternative for WPF, which can fully utilize their respective advantages [74, 77].

This chapter proposes a novel ensemble method for WPF, which integrates feedforward neural networks, wavelet transform, input feature selection as well as partial least squares regression. The three-layer feedforward neural networks with Levenberg-Marquardt (LM) learning algorithm are used as the forecasting engine. The feature selection method CMIFS in Chapter 4 is used to select a set of input variables for the WPF model. In order to overcome the nonstationarity of wind power series and improve the forecasting performance, the wavelet based ensemble scheme in Chapter 4 is integrated into the WPF model. The individual outputs are combined to form the ensemble forecast by PLSR. In order to confirm the effectiveness, the proposed method is tested on real-world datasets and compared to other forecasting methods.

5.2 Proposed Wind Power Forecasting Method

The structure of the proposed ensemble method is illustrated in Figure 5.1. Only the wind power and wind speed measurements are used as input variables to the forecast model. The wind power is the generation output of a wind farm and the wind speed is measured at the site of the wind farm. Other data such as

wind direction and temperature are not utilized here, but can be easily included in future work. Therefore, a candidate set of input features, denoted by $\Gamma(t)$, is given by

$$\Gamma(t) = \{\mathrm{WP}(t-1),...,\mathrm{WP}(t-100),\mathrm{WS}(t),...,\mathrm{WS}(t-100)\} \qquad (5.1)$$

where t is the time index. Since hourly wind power forecasting is studied in this chapter, t is on an hourly basis. WP(t-1),…,WP(t-100) and WS(t),…,WS(t-100) are the wind power and speed data up to 100 hours ago, respectively. Since the wind power data does not exhibit apparent recurring trends, the order of past values is set to be 100, which is sufficient to contain all useful features. Similar theory is applied to the wind speed data. The candidate set $\Gamma(t)$ totally has 201 features, which cannot be directly used in WPF. The CMIFS method proposed in Chapter 4 is therefore used to carry out relevance and redundancy analysis on $\Gamma(t)$ and select a small set of input variables. In the literature, the number of inputs used for WPF is at most twenty or thirty. Thus, the threshold T is set to be forty in this chapter, which is deemed enough to include all relevant features.

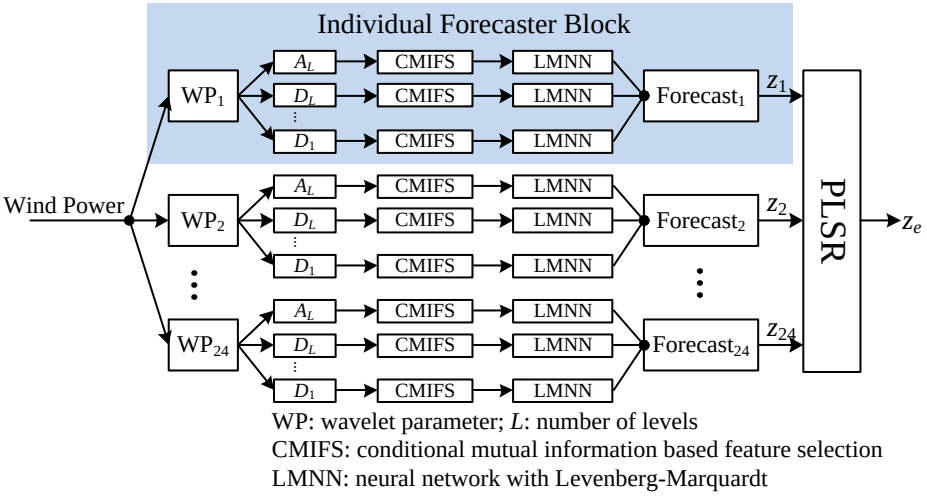


Figure 5.1 Proposed ensemble model for WPF.

In this chapter, the wavelet transform is used to decompose the wind power data into a set of constitutive components that have different frequencies. Then each constitutive component is forecasted by a three-layer feedforward neural network with the Levenberg-Marquardt learning method (LMNN). In order to improve the generalization capability and avoid overtraining, the early stopping criterion [45] and the Nguyen-Widrow initialization technique [157] are used in LMNN.

The wavelet-based ensemble model is featured with different mixtures of the mother wavelet and the number of decomposition levels. In order to select appropriate wavelet specifications, the attributes of mother wavelet and the characteristics of wind power data must be taken into account [107]. Daubechies (db), Coiflets (coif) and Symlets (sym) are compactly supported wavelets with high number of vanishing moments [111]. These properties can reach an appropriate trade-off between wavelength and smoothness, which are very suitable for treating the nonstationary wind power series [78]. In fact, the Daubechies wavelets have been widely used to decompose the wind power data [75, 78]. In this chapter, we expand to two more wavelet families: Coiflets and Symlets. The order of mother wavelet used is from 2 to 5 and the number of decomposition levels is 1 and 2. Hence, 12 mother wavelets (db2–db5, coif2–coif5 and sym2–sym5) are employed and the proposed wavelet-based ensemble strategy contains 24 sets of wavelet parameters. It is noted that more individual forecasters can be included by increasing the number of decomposition levels. Following this scheme, the diversity of ensemble will be amplified and better forecasting performance can be expected. However, the computational time

would increase dramatically.

The proposed ensemble method consists of 24 individual forecaster blocks. Each forecaster block is made up of wavelet transform, input feature selection and three-layer feedforward neural networks (LMNN). In each forecaster block, the CMIFS technique is applied to the constitutive components (A_L, D_L,…, D_1). Taking A_L as an example, the candidate input set is {$A_L(t\text{-}1)$,…, $A_L(t\text{-}100)$, WS(t),…, WS(t-100)}. For each constitutive component, a three-layer LMNN is used to predict the hourly wind power output of the next day. The Levenberg-Marquardt learning algorithm is used to train the neural networks. Finally, the 24 individual forecasts (z_1, z_2,…, z_{24}) are aggregated to generate the ensemble output (z_e), where the weight factors are determined by the PLSR method.

5.3 Results

In this section, the proposed method was tested using the data collected by the National Renewable Energy Laboratory [19]. Two different wind farms were considered for testing. The first wind farm at site 1 is located in latitude 34.98° and longitude –104.04°, with installed capacity of 171.8 MW. The second wind farm at site 50 is located in latitude 47.58° and longitude –97.5°, with installed capacity of 1003.7 MW. It is clear that the two wind farms are quite different in site location and installed capacity. The 10-minute wind speed and wind power measurements from 2005 to 2006 of two wind farms were used for the study. The six readings over an hour were averaged to produce hourly data. The site 1 wind farm data were used in Cases 1 to 5, while the site 50 wind farm data were used in Case 6.

In order to evaluate the performance, two error measures: mean absolute percentage error (MAPE) and normalized root mean square error (NRMSE) were calculated by

$$\begin{aligned} \text{MAPE} &= \frac{100\%}{M}\sum_{i=1}^{M}\left|\frac{y_i-\hat{y}_i}{y_c}\right| \\ \text{NRMSE} &= \frac{100\%}{y_c}\sqrt{\frac{1}{M}\sum_{i=1}^{M}\left(y_i-\hat{y}_i\right)^2} \end{aligned} \tag{5.2}$$

where M is the number of hours in the testing period, y_i is the actual wind power value, $\hat{y}_i$ is the forecasted wind power value and y_c is the installed capacity of wind farm.

a) Case 1

It is pointed out that any advanced model for WPF should first be compared with the persistence method [60], which has been presented in Section 2.3.3. The persistence method performs well for very short-term forecasts but its error grows with the increasing forecast horizon. In order to obtain a good performance over a longer horizon, a new reference model was presented in [158], which is given by

$$\text{WP}(t+\Delta t) = a\,\text{WP}(t) + (1-a)\,\text{WPA}(t) \tag{5.3}$$

where WP(t) is the wind power at time t and Δt is the forecast horizon, a is the correlation factor between WP(t) and WP(t+Δt), and WPA(t) is the average of the past wind power values prior to t. The two quantities WPA(t) and a are estimated by [158]

$$\text{WPA}(t) = \frac{1}{N_{\text{WP}}}\sum_{t=1}^{N_{\text{WP}}}\text{WP}(t) \tag{5.4}$$

$$a = \frac{\sum_{t=1}^{N_{WP}} \mathrm{WPX}(t)\mathrm{WPX}(t+\Delta t)}{\sqrt{\sum_{t=1}^{N_{WP}} \mathrm{WPX}(t)^2}\sqrt{\sum_{t=1}^{N_{WP}} \mathrm{WPX}(t+\Delta t)^2}} \tag{5.5}$$

and

$$\mathrm{WPX}(t) = \mathrm{WP}(t) - \mathrm{WPA}(t) \tag{5.6}$$

where N_{WP} is the number of past wind power measurements used in the calculation.

In this case, the proposed method was compared to the above two reference models using the data from the site 1 wind farm. The training set is from January 1, 2005 to December 31, 2005 and the testing set is from January 1, 2006 to December 31, 2006. Forty percent of data in the training set were used for validation.

The wind speeds used in the simulation are actual measured data. For each constitutive component, the number of selected input features and the number of hidden nodes were selected through extensive tests. All simulations were run in Matlab on a computer with a 2.66-GHz CPU. Note that the forecast model is not retrained in the entire testing period, but more retraining may be a good attempt to improve the forecasting accuracy.

The 1-hour to 48-hour ahead forecasting results are shown in Table 5.1. It is seen that the new reference model obtains better forecasting performance than the persistence model in 24- and 48-hour cases. The percentage increment of the proposed method with respect to the two

reference models is shown in Figure 5.2. For the given testing period, the proposed method presents significant improvements over the other two reference methods in all look-ahead times. The proposed method is 82.5% and 77.9% better than the persistence and new reference models respectively in the whole range of 48 look-ahead hours in MAPE. Similar improvements have also been seen in the measure of NRMSE.

Table 5.1 Forecasting results of persistence, new reference and the proposed method

		1-hour	24-hour	48-hour
Persistence	MAPE	7.83	26.90	30.67
	NRMSE	12.50	35.50	39.21
New Reference	MAPE	8.30	21.48	24.37
	NRMSE	12.26	23.86	26.93
Proposed	MAPE	0.49	4.83	5.38
	NRMSE	0.66	6.29	6.98

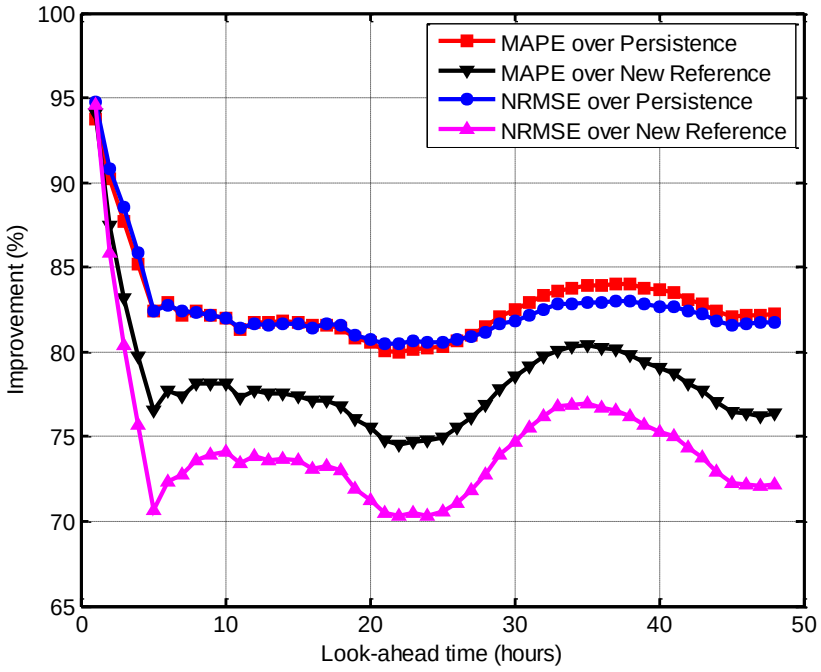


Figure 5.2 Improvement of proposed method over persistence and new reference.

b) Case 2

The proposed method is compared to three models: LMNN, radial basis function neural network (RBFNN) and wavelet neural network (WNN) using the data in Case 1. The combination of mother wavelet coif4 and 2-level decomposition is used for WNN, which can obtain the best average forecasting accuracy for 48 look-ahead hours. The MAPE and NRMSE results of the four models are given in Figures 5.3 and 5.4, respectively.

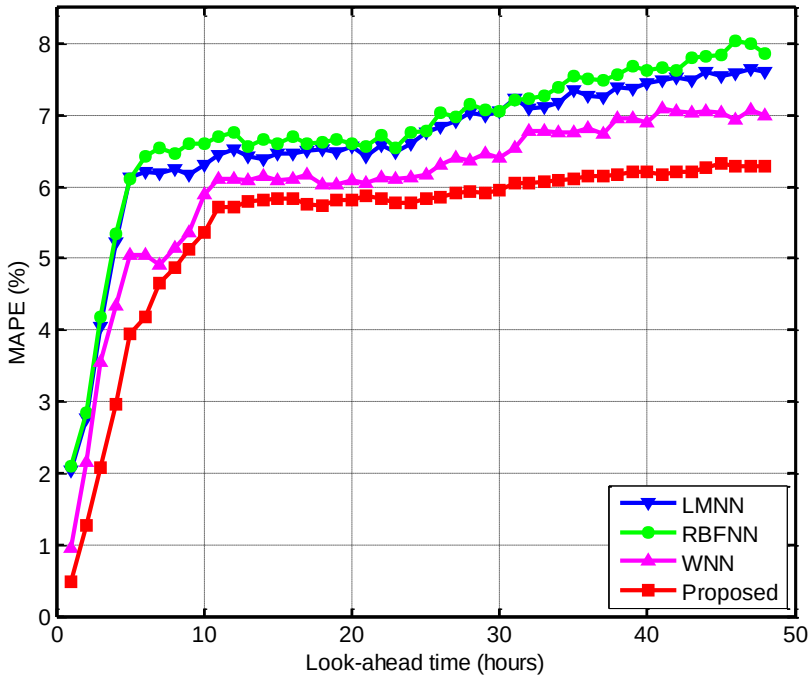


Figure 5.3 MAPE results of LMNN, RBFNN, WNN and proposed method.

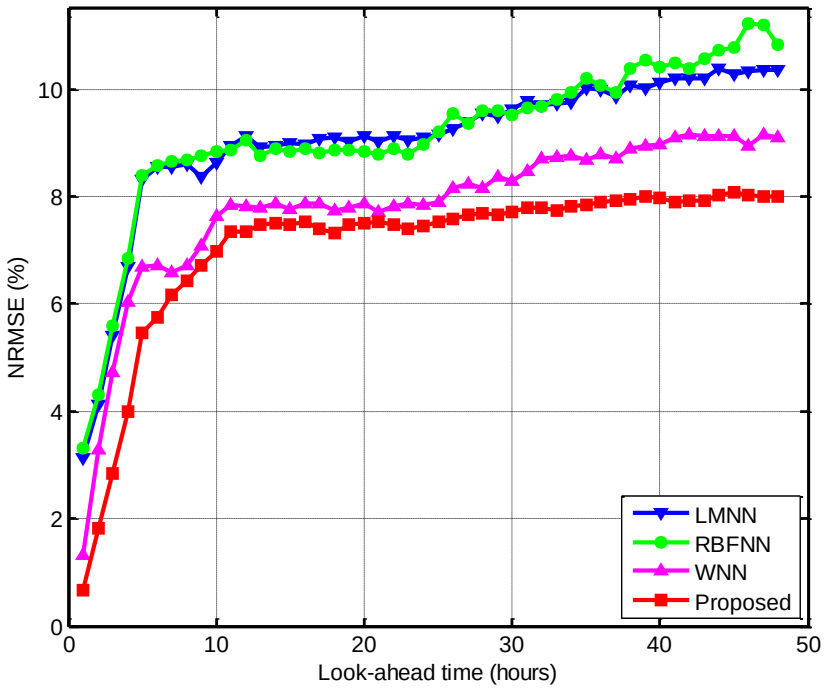


Figure 5.4 NRMSE results of LMNN, RBFNN, WNN and proposed method.

Firstly, it is seen that the errors of all methods grow dramatically in the first few look-ahead hours. Secondly, the two models LMNN and RBFNN achieve similar results, especially in short horizons. Thirdly, the forecasting accuracy is clearly improved when wavelet transform is involved. Compared to LMNN, the average accuracy of WNN is 8.48% and 13.6% better in MAPE and NRMSE, respectively. Fourthly, it is clearly shown that the proposed ensemble strategy is an effective tool to improve the forecasting accuracy. For example, the proposed ensemble model has yielded an average drop of 10.9% in MAPE with respect to WNN. Finally, the proposed method outperforms other alternatives in all look-ahead times, which confirms its effectiveness and superiority.

c) Case 3

This case compares different feature selection methods for the data used in Case 1. For each sub-component (A_L, D_L,…, D_1), different feature selection methods are used to select the same number of inputs. The results for 1-hour ahead forecasting are shown in Table 5.2. The benchmark methods in Table 5.2 include correlation analysis (CA), mutual information (MI), RReliefF [153] and minimum redundancy maximum relevance (mRMR) [159]. The first three methods choose the features with top *scores* (e.g., correlation coefficient for CA) from the candidate input set and do not consider the redundancy problem. The mRMR method regulates the feature relevance and redundancy by comparing the new feature with the selected ones, like the proposed CMIFS method.

Table 5.2 1-hour ahead forecasting results of five feature selection methods

	MAPE	NRMSE
CA	0.58	0.83
MI	0.62	0.91
RReliefF	0.61	0.87
mRMR	1.51	2.18
CMIFS	0.49	0.66

As clearly shown in Table 5.2, the best forecasting performance for the testing period is obtained by the proposed CMIFS method. For example, CMIFS can raise the forecasting accuracy by 15.5%, 21.0%, 19.7% and 67.5% in MAPE with respect to the previous four methods, respectively. The improvements over CA, MI and RReliefF imply that relevance analysis alone is insufficient for feature selection. However, the mRMR method including redundancy analysis also produces poor

results. The reason could be that the redundancy test is too stringent, which removes too many strongly relevant input features.

d) Case 4

The forecasting performance of individual forecast and ensemble forecast is further studied in this case. The individual forecast employs a set of wavelet parameters while the ensemble forecast assembles the individual forecasts using PLSR. The forecasting results (in MAPE) of four different look-ahead times (1-, 6-, 12- and 24-hour ahead) are tabulated in Table 5.3.

Table 5.3 Individual and ensemble forecasting results in MAPE of four different look-ahead times

Level	Wavelet	1-hour	6-hour	12-hour	24-hour
1	coif2	1.06	5.17	5.95	6.06
	coif3	1.38	4.61	5.94	5.94
	coif4	1.18	4.79	5.98	5.98
	coif5	1.57	5.14	6.06	6.10
	sym2	2.96	4.52	6.07	6.00
	sym3	2.67	4.60	5.93	5.93
	sym4	2.09	4.68	5.97	6.04
	sym5	1.39	4.52	5.95	5.99
	db2	2.90	4.62	5.94	5.97
	db3	2.56	4.56	5.98	5.95
	db4	2.34	4.54	6.03	5.94
	db5	1.63	4.53	5.96	6.05
2	coif2	1.06	4.95	6.13	6.11
	coif3	0.86	4.94	6.03	6.28
	coif4	0.96	5.05	6.10	6.13
	coif5	0.87	5.07	6.19	6.37
	sym2	1.01	5.28	6.37	6.24
	sym3	1.74	5.23	6.10	6.33
	sym4	0.98	4.93	6.19	6.17
	sym5	0.81	4.94	6.25	6.35
	db2	1.01	5.29	6.56	6.63
	db3	1.61	4.90	6.21	6.23
	db4	1.49	4.84	6.14	6.12
	db5	0.88	5.00	6.15	6.15
LAST12		0.57	4.43	5.81	5.89
ALL24		0.49	4.19	5.71	5.78

In Table 5.3, “LAST12” refers to the ensemble forecast which only merges the individual forecasts using 2-level decomposition. “ALL24” is the proposed method using all the 24 individual forecasts. It is seen that different wavelet parameters result in different forecast results for a certain look-ahead time. A specific set of wavelet parameters cannot obtain the best result in all look-ahead times. Moreover, the two ensemble forecasts (LAST12 and ALL24) have provided better results than any of the individual forecasts, which confirms the effectiveness of the proposed wavelet-based ensemble scheme. In addition, compared to LAST12, ALL24 is able to produce better results even with some *bad* individual forecasts (i.e., with large MAPEs). This implies that the *bad* individuals still contain some independent information that contributes to the ensemble forecast accuracy.

e) Case 5

The preceding four case studies are made based on the measured wind speed data. In practice, the forecasted wind speed values should be included in WPF, which are provided by the near weather station. In this case, the effect of wind speed forecasting errors on WPF is studied. A noisy wind speed data series is used to simulate the prediction errors. The Gaussian noise of zero mean and standard deviation of 0.5 m/s is added to the measured wind speed series. The simulated wind speed forecasting error fluctuates between -2.16 m/s and 2.21 m/s. The 1-hour to 24-hour ahead prediction results using measured and noisy wind speed data are shown in Figure 5.5.

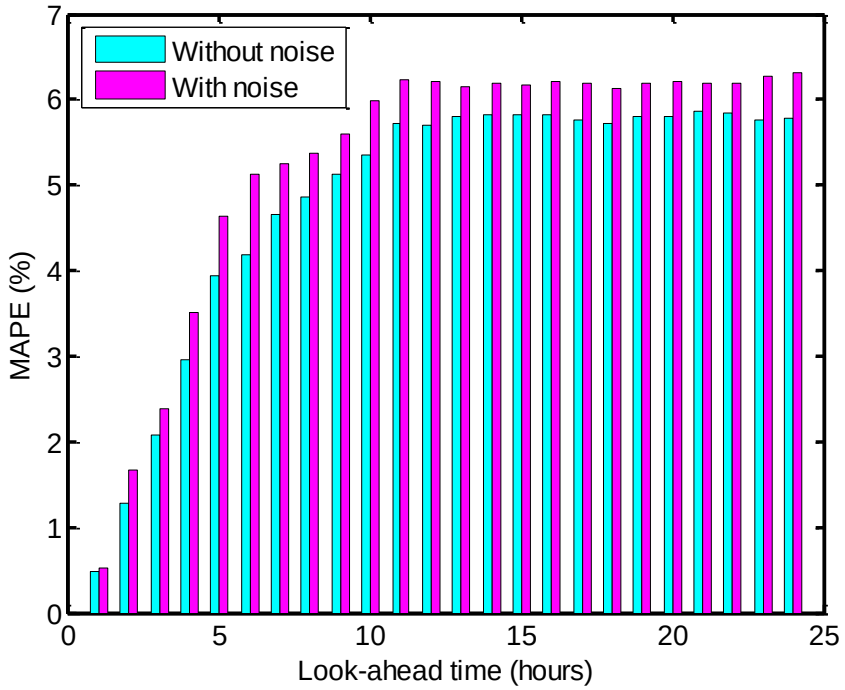


Figure 5.5 1-hour to 24-hour ahead forecasting results using measured and noisy wind speed data.

As shown in Figure 5.5, the proposed method is able to produce encouraging forecasting results using the noisy wind speed data. The average forecasting error increases about 9.22% over the whole range of 24 look-ahead hours. Since the wind power series does not present long-term trends, the input variables related to wind power are always the ones with small lagged times. The quality of wind speed data, in this situation, would have a great impact on the multi-step WPF.

f) Case 6

The proposed method is tested using the site 50 wind farm data. It should be pointed out that the wind farm at site 50 is quite different in installed capacity and location from the one at site 1. The persistence

and new reference methods in Case 1 and the LMNN, RBFNN and WNN methods in Case 2 are involved for the purpose of comparison. The training, validation and testing periods are defined as the same as in Case 1. The forecasting results of 1-, 24- and 48-hour cases are shown in Table 5.4.

Table 5.4 Forecasting results for the models in Case 6

		1-hour	24-hour	48-hour
Persistence	MAPE	7.66	24.62	28.37
	NRMSE	11.78	31.90	36.01
New Reference	MAPE	7.91	16.43	20.62
	NRMSE	11.72	19.14	23.83
LMNN	MAPE	1.35	5.98	6.48
	NRMSE	1.92	7.63	8.28
RBFNN	MAPE	1.56	6.09	6.60
	NRMSE	2.25	7.70	8.45
WNN	MAPE	0.86	5.16	5.78
	NRMSE	1.17	6.74	7.54
Proposed	MAPE	0.44	4.44	5.07
	NRMSE	0.59	5.80	6.59

It can be observed that the proposed method has provided better forecasting performance than other models in all testing cases, which verifies its feasibility and effectiveness. Taking the 48-hour case as an example, the increases in MAPE of the proposed method with respect to the previous five methods are 82.1%, 75.4%, 21.8%, 23.2% and 12.3%, respectively.

5.4 Conclusion

This chapter proposes an ensemble model for WPF, which contains neural networks, wavelet transform, input variable selection and partial least squares regression. An innovative ensemble scheme using different wavelet parameters

is proposed to construct the neural network ensemble model. A feature selection technique called CMIFS is developed to select the most informative input data for the forecasting model. In order to obtain an accurate and reliable ensemble forecast, PLSR is employed to aggregate the individual forecasts.

In order to demonstrate the effectiveness, the proposed method has been tested on actual data from National Renewable Energy Laboratory. Compared to other forecasting methods, the proposed method can produce better results in the whole range of look-ahead times. The improved forecasting accuracy can be attributed to several factors, like the proposed wavelet-based ensemble scheme, the proposed feature selection technique, the early stopping criterion and the partial least squares regression method.

The proposed approach for WPF presents many advantages. Firstly, it can overcome the difficulty induced by the nonstationarity of wind power series by wavelet transform. Secondly, it can alleviate many trivial problems in WPF such as random network weights and biases, wavelet parameter determination. Thirdly, as clearly shown in Case 5, it has robustness with respect to large wind speed forecasting errors. Finally, it is able to provide high quality forecasting results for wind farms with different locations and sizes, which has been shown in Case 6.

Chapter 6 Conclusions and Future Works

6.1 Conclusions

This thesis discusses the importance of energy forecasting for the operation and planning of electric power systems. It is always an integral part of energy management system and assists an electric utility operator to make important decisions such as unit commitment and load switching. It becomes increasingly important due to the significant changes taking place in modern power systems, like smart grid technologies, renewable energy and energy market deregulation.

Forecasting is never 100% correct, which drives us to put continuous effort in making the forecasting as accurate as possible. In this thesis, the research focus is on developing advanced methods for load and wind power forecasting. The relevant conventional and state-of-the-art methodologies are reviewed. It is found that load forecasting and wind power forecasting have similarities as well as differences. The load data and wind power data are both nonstationary. The load data usually present trends and periodicities, while the wind power data are extremely chaotic. Some methods of load forecasting such as linear regression, time series, wavelet transform and neural networks can also be utilized for wind power forecasting. Some methods are used for a single purpose like numerical weather prediction for wind power forecasting. This thesis proposes two novel

techniques for short-term load forecasting and one hybrid approach for wind power forecasting.

Firstly, a novel short-term load forecasting approach is proposed based on wavelet transform, ELM and MABC algorithm. The wavelet transform is used to decompose the load series for capturing the complicated features at different frequencies. Each component of the load series is then separately forecasted by a hybrid predictor consisting of ELM and MABC. The global search technique MABC is developed to look for the best parameters of input weights and hidden biases for ELM. Compared to the conventional neuro-evolution method, ELM-MABC can improve the learning accuracy with fewer iteration steps.

Secondly, a novel ELM-based ensemble method is proposed for short-term load forecasting, which includes wavelet-based ensemble strategy, input feature selection, hybrid neural networks as well as partial least squares regression. The wavelet transform is used not only to decompose the load data but also to create the individual ELM-based forecasters. In order to boost the learning accuracy of SLFNs, a hybrid learning algorithm combining ELM and Levenberg-Marquardt is developed. A new feature selection method based on the conditional mutual information is derived to select a small set of input variables for the forecasting model. Finally, to make an accurate and reliable ensemble forecast, partial least squares regression is used as the combining method to aggregate the individual forecasts.

Thirdly, a novel ensemble method consisting of neural networks, wavelet-based ensemble strategy, feature selection and partial least squares regression is proposed for the generation forecasting of a wind farm. The embedded neural

networks are trained using the popular Levenberg-Marquardt leaning algorithm. The proposed wavelet-based ensemble scheme, feature selection technique and partial least squares regression method are fine-tuned to meet the requirements of wind power forecasting. The proposed method is used to carry out 1-hour to 48-hour ahead forecasting using the data from two different wind farms.

In order to confirm their effectiveness, the proposed methods were tested using actual electrical load and wind power data. Numerical results reveal that the proposed methods can provide better forecasting performance than other standard and state-of-the-art methods. The proposed methods present high potential for practical forecasting applications in power systems.

6.2 Future Works

Some recommendations for future research are as follows:

1) The research study addressing wind ramp forecasting is limited. Ramp events are the sudden and large fluctuations in wind power production. Two parameters are used to describe the ramp: the size of ramp and the duration time of ramp. Ramp events threaten the security of power grids. For example, for a ramp-down event, the wind power decreases rapidly. The operators need to turn on the spinning reserves in a short time (few minutes) to fill the demand gap. Therefore, it is necessary to develop an accurate wind ramp forecasting system.
2) In this thesis, only the point forecasting problem is investigated, i.e., all the proposed methods finally obtain either a future load or wind power value. However, a single-valued point forecast may not be sufficient in

practical applications [87]. In some cases, the uncertainties behind the point forecasts are also needed. Probabilistic forecasting, which creates a prediction interval with a certain probability, can present more useful information for the decision-making processes [87, 88]. Hence, we can extend our deterministic methods to probabilistic load and wind power forecasting.

3) The proposed energy forecasting methods have shown a great potential to be applied in real-world applications. A part of research focus can be shifted to inspect the influences of the proposed forecasting methods on spinning reserves determination, demand response management, energy purchasing, power system reliability, etc.

Author's Publications

1. **S. Li**, P. Wang and L. Goel, "Electric load forecasting using wavelet transform and extreme learning machine", in *Proceedings of 22th European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning (ESANN 2014)*, pp. 631-636, **2014**.
2. **S. Li**, P. Wang and L. Goel, "Short-term load forecasting by wavelet transform and evolutionary extreme learning machine", *Electric Power Systems Research*, vol. 122, pp. 96-103, **2015**.
3. **S. Li**, P. Wang and L. Goel, "A novel wavelet-based ensemble method for short-term load forecasting with hybrid neural networks and feature selection", *Power Systems*, *IEEE Transactions on,* **Accepted**.
4. **S. Li**, P. Wang and L. Goel, "Wind power forecasting using neural network ensembles with feature selection", *Sustainable Energy*, *IEEE Transactions on,* vol. 6, pp. 1447-1456, **2015**.
5. **S. Li**, L. Goel and P. Wang, "An ensemble approach for short-term load forecasting by extreme learning machine", *Applied Energy*, Submitted.

Bibliography

[1] *International Energy Outlook 2013*. Available: http://www.eia.gov/forecasts/archive/aeo13/

[2] S. M. Amin and B. F. Wollenberg, "Toward a smart grid: power delivery for the 21st century," *Power and Energy Magazine, IEEE,* vol. 3, pp. 34-41, 2005.

[3] H. Farhangi, "The path of the smart grid," *Power and Energy Magazine, IEEE,* vol. 8, pp. 18-28, 2010.

[4] J. D. Liu, T. T. Lie, and K. L. Lo, "An empirical method of dynamic oligopoly behavior analysis in electricity markets," *Power Systems, IEEE Transactions on,* vol. 21, pp. 499-506, 2006.

[5] M. Parvania, M. Fotuhi-Firuzabad, and M. Shahidehpour, "Optimal demand response aggregation in wholesale electricity markets," *Smart Grid, IEEE Transactions on,* vol. 4, pp. 1957-1965, 2013.

[6] *Renewables 2013 Global Status Report*. Available: http://www.ren21.net/Portals/0/documents/Resources/GSR/2013/GSR2013_lowres.pdf

[7] *The First Decade: 2004-2014*. Available: http://www.ren21.net/Portals/0/documents/activities/Topical%20Reports/REN21_10yr.pdf

[8] C. Bishop, *Neural Networks for Pattern Recognition*. Oxford: Clarendon Press, 1995.

[9] G. Gross and F. D. Galiana, "Short-term load forecasting," *Proceedings of the IEEE,* vol. 75, pp. 1558-1573, 1987.

[10] H. L. Willis, *Spatial Electric Load Forecasting*. New York: Marcel Dekker, 1996.

[11] J. Jung and R. P. Broadwater, "Current status and future advances for wind speed and power forecasting," *Renewable and Sustainable Energy*

Reviews, vol. 31, pp. 762-777, 2014.

[12] H. S. Hippert, C. E. Pedreira, and R. C. Souza, "Neural networks for short-term load forecasting: a review and evaluation," *Power Systems, IEEE Transactions on,* vol. 16, pp. 44-55, 2001.

[13] S. A. Soliman and A. M. Alkandari, *Electrical Load Forecasting: Modeling and Model Construction*. Burlington, MA: Butterworth-Heinemann, 2010.

[14] E. A. Feinberg and D. Genethliou, *Applied Mathematics for Restructured Electric Power Systems: Optimization, Control, and Computational Intelligence: Load Forecasting*. New York: Springer, 2005.

[15] C. E. Asbury, "Weather load model for electric demand and energy forecasting," *Power Apparatus and Systems, IEEE Transactions on,* vol. 94, pp. 1111-1116, 1975.

[16] *ISO New England Data*. Available: http://www.iso-ne.com/isoexpress/web/reports/pricing/-/tree/zone-info

[17] S. Fan, K. Methaprayoon, and W. J. Lee, "Multiregion load forecasting for system with large geographical area," *Industry Applications, IEEE Transactions on,* vol. 45, pp. 1452-1459, 2009.

[18] M. T. Hagan and S. M. Behr, "The time series approach to short term load forecasting," *Power Systems, IEEE Transactions on,* vol. 2, pp. 785-791, 1987.

[19] *National Renewable Energy Laboratory Data*. Available: http://www.nrel.gov/electricity/transmission/eastern_wind_methodology.html

[20] T. Senjyu, H. Takara, K. Uezato, and T. Funabashi, "One-hour-ahead load forecasting using neural network," *Power Systems, IEEE Transactions on,* vol. 17, pp. 113-118, 2002.

[21] H. M. Kutner, J. C. Nachtsheim, J. Neter, and W. Li, *Applied Linear Statistical Models*, 5th ed. Boston: McGraw-Hill/Irwin, 2004.

[22] T. Hong, P. Wang, and H. L. Willis, "A naive multiple linear regression benchmark for short term load forecasting," in *Proceedings of the 2011*

IEEE Power and Energy Society General Meeting, 2011, pp. 1-6.

[23] R. F. Engle, C. Mustafa, and J. Rice, "Modelling peak electricity demand," *Journal of Forecasting,* vol. 11, pp. 241-251, 1992.

[24] T. Haida and S. Muto, "Regression based peak load forecasting using a transformation technique," *Power Systems, IEEE Transactions on,* vol. 9, pp. 1788-1794, 1994.

[25] G. E. P. Box, G. M. Jenkins, and G. C. Reinsel, *Time Series Analysis: Forecasting and Control*. Hoboken, NJ: Wiley, 2008.

[26] N. Amjady, "Short-term hourly load forecasting using time-series modeling with peak load estimation capability," *Power Systems, IEEE Transactions on,* vol. 16, pp. 798-805, 2001.

[27] H. T. Yang and C. M. Huang, "A new short-term load forecasting approach using self-organizing fuzzy ARMAX models," *Power Systems, IEEE Transactions on,* vol. 13, pp. 217-225, 1998.

[28] J. Y. Fan and J. D. McDonald, "A real-time implementation of short-term load forecasting for distribution power systems," *Power Systems, IEEE Transactions on,* vol. 9, pp. 988-994, 1994.

[29] C. M. Huang, C. J. Huang, and M. L. Wang, "A particle swarm optimization to identifying the ARMAX model for short-term load forecasting," *Power Systems, IEEE Transactions on,* vol. 20, pp. 1126-1133, 2005.

[30] K. Metaxiotis, A. Kagiannas, D. Askounis, and J. Psarras, "Artificial intelligence in short term electric load forecasting: a state-of-the-art survey for the researcher," *Energy Conversion and Management,* vol. 44, pp. 1525-1534, 2003.

[31] S. S. Haykin, *Neural Networks: A Comprehensive Foundation*, 2nd ed. Upper Saddle River, NJ: Prentice Hall, 1999.

[32] S. T. Chen, D. C. Yu, and A. R. Moghaddamjo, "Weather sensitive short-term load forecasting using nonfully connected artificial neural network," *Power Systems, IEEE Transactions on,* vol. 7, pp. 1098-1105, 1992.

[33] J. Vermaak and E. C. Botha, "Recurrent neural networks for short-term

load forecasting," *Power Systems, IEEE Transactions on,* vol. 13, pp. 126-132, 1998.

[34] C. N. Lu, H. T. Wu, and S. Vemuri, "Neural network based short term load forecasting," *Power Systems, IEEE Transactions on,* vol. 8, pp. 336-342, 1993.

[35] H. Yoo and R. L. Pimmely, "Short term load forecasting using a self-supervised adaptive neural network," *Power Systems, IEEE Transactions on,* vol. 14, pp. 779-784, 1999.

[36] K. L. Ho, Y. Y. Hsu, and C. C. Yang, "Short term load forecasting using a multilayer neural network with an adaptive learning algorithm," *Power Systems, IEEE Transactions on,* vol. 7, pp. 141-149, 1992.

[37] A. S. Weigend and A. N. Srivastava, "Predicting conditional probability distributions: a connectionist approach," *International Journal of Neural Systems,* vol. 06, pp. 109-118, 1995.

[38] A. Khosravi and S. Nahavandi, "Combined nonparametric prediction intervals for wind power generation," *Sustainable Energy, IEEE Transactions on,* vol. 4, pp. 849-856, 2013.

[39] P. Jackson, *Introduction to Expert Systems*. Harlow, England: Addison-Wesley, 1999.

[40] K. L. Ho, Y. Y. Hsu, C. F. Chen, T. E. Lee, C. C. Liang, T. S. Lai*, et al.*, "Short term load forecasting of Taiwan power system using a knowledge-based expert system," *Power Systems, IEEE Transactions on,* vol. 5, pp. 1214-1221, 1990.

[41] S. Rahman and O. Hazim, "Load forecasting for multiple sites: development of an expert system-based technique," *Electric Power Systems Research,* vol. 39, pp. 161-169, 1996.

[42] M. Tamimi and R. Egbert, "Short term electric load forecasting via fuzzy neural collaboration," *Electric Power Systems Research,* vol. 56, pp. 243-248, 2000.

[43] C. Cortes and V. Vapnik, "Support-vector networks," *Machine Learning,* vol. 20, pp. 273-297, 1995.

[44] K. Muller, S. Mika, G. Ratsch, K. Tsuda, and B. Scholkopf, "An

introduction to kernel-based learning algorithms," *Neural Networks, IEEE Transactions on,* vol. 12, pp. 181-201, 2001.

[45] S. S. Haykin, *Neural Networks and Learning Machines*, 3rd ed. New York: Prentice Hall, 2009.

[46] M. Mohandes, "Support vector machines for short-term electrical load forecasting," *International Journal of Energy Research,* vol. 26, pp. 335-345, 2002.

[47] Y. Wang, Q. Xia, and C. Kang, "Secondary forecasting based on deviation analysis for short-term load forecasting," *Power Systems, IEEE Transactions on,* vol. 26, pp. 500-507, 2011.

[48] S. Fan and L. Chen, "Short-term load forecasting based on an adaptive hybrid method," *Power Systems, IEEE Transactions on,* vol. 21, pp. 392-401, 2006.

[49] H. Mori and H. Kobayashi, "Optimal fuzzy inference for short-term load forecasting," *Power Systems, IEEE Transactions on,* vol. 11, pp. 390-396, 1996.

[50] J. H. Park, Y. M. Park, and K. Y. Lee, "Composite modeling for adaptive short-term load forecasting," *Power Systems, IEEE Transactions on,* vol. 6, pp. 450-457, 1991.

[51] Z. Yu, "A temperature match based optimization method for daily load prediction considering DLC effect," *Power Systems, IEEE Transactions on,* vol. 11, pp. 728-733, 1996.

[52] M. Alamaniotis, A. Ikonomopoulos, and L. H. Tsoukalas, "Evolutionary multiobjective optimization of kernel-based very short-term load forecasting," *Power Systems, IEEE Transactions on,* vol. 27, pp. 1477-1484, 2012.

[53] V. H. Hinojosa and A. Hoese, "Short-term load forecasting using fuzzy inductive reasoning and evolutionary algorithms," *Power Systems, IEEE Transactions on,* vol. 25, pp. 565-574, 2010.

[54] H. T. Yang, C. M. Huang, and C. L. Huang, "Identification of ARMAX model for short term load forecasting: an evolutionary programming approach," *Power Systems, IEEE Transactions on,* vol. 11, pp. 403-408,

1996.

[55] N. Amjady, F. Keynia, and H. Zareipour, "Short-term load forecast of microgrids by a new bilevel prediction strategy," *Smart Grid, IEEE Transactions on,* vol. 1, pp. 286-294, 2010.

[56] I. Drezga and S. Rahman, "Input variable selection for ANN-based short-term load forecasting," *Power Systems, IEEE Transactions on,* vol. 13, pp. 1238-1244, 1998.

[57] *Global Wind Energy Outlook 2014*. Brussels, Belgium: Global Wind Energy Council, 2014.

[58] S. Fan, J. R. Liao, R. Yokoyama, L. Chen, and W. J. Lee, "Forecasting the wind generation using a two-stage network based on meteorological information," *Energy Conversion, IEEE Transactions on,* vol. 24, pp. 474-482, 2009.

[59] A. M. Foley, P. G. Leahy, A. Marvuglia, and E. J. McKeogh, "Current methods and advances in forecasting of wind power generation," *Renewable Energy,* vol. 37, pp. 1-8, 2012.

[60] S. S. Soman, H. Zareipour, O. Malik, and P. Mandal, "A review of wind power and wind speed forecasting methods with different time horizons," in *North American Power Symposium*, 2010, pp. 1-8.

[61] G. M. Masters, *Renewable and Efficient Electric Power Systems*. Hoboken, NJ: John Wiley & Sons, 2004.

[62] L. Ma, S. Luan, C. Jiang, H. Liu, and Y. Zhang, "A review on the forecasting of wind speed and generated power," *Renewable and Sustainable Energy Reviews,* vol. 13, pp. 915-920, 2009.

[63] *Vestas V110-2.0 MW*. Available: http://www.vestas.com/en/products_and_services/turbines/v110-2_0_mw

[64] L. Landberg, "Short-term prediction of the power production from wind farms," *Journal of Wind Engineering and Industrial Aerodynamics,* vol. 80, pp. 207-220, 1999.

[65] U. Focken, M. Lange, and H. P. Waldl, "Previento - A wind power prediction system with an innovative upscaling algorithm," in

Proceedings of the 2001 European Wind Energy Conference & Exhibition, 2001.

[66] C. W. Potter and M. Negnevitsky, "Very short-term wind forecasting for Tasmanian power generation," *Power Systems, IEEE Transactions on,* vol. 21, pp. 965-972, 2006.

[67] R. G. Kavasseri and K. Seetharaman, "Day-ahead wind speed forecasting using f-ARIMA models," *Renewable Energy,* vol. 34, pp. 1388-1393, 2009.

[68] H. Liu, H. Q. Tian, C. Chen, and Y. F. Li, "A hybrid statistical method to predict wind speed and wind power," *Renewable Energy,* vol. 35, pp. 1857-1861, 2010.

[69] N. Amjady, F. Keynia, and H. Zareipour, "Wind power prediction by a new forecast engine composed of modified hybrid neural network and enhanced particle swarm optimization," *Sustainable Energy, IEEE Transactions on,* vol. 2, pp. 265-276, 2011.

[70] K. Bhaskar and S. N. Singh, "AWNN - Assisted wind power forecasting using feed-forward neural network," *Sustainable Energy, IEEE Transactions on,* vol. 3, pp. 306-315, 2012.

[71] N. Amjady, F. Keynia, and H. Zareipour, "Short-term wind power forecasting using ridgelet neural network," *Electric Power Systems Research,* vol. 81, pp. 2099-2107, 2011.

[72] G. Li and J. Shi, "On comparing three artificial neural networks for wind speed forecasting," *Applied Energy,* vol. 87, pp. 2313-2320, 2010.

[73] E. Mangalova and E. Agafonov, "Wind power forecasting using the k-nearest neighbors algorithm," *International Journal of Forecasting,* vol. 30, pp. 402-406, 2014.

[74] I. G. Damousis, M. C. Alexiadis, J. B. Theocharis, and P. S. Dokopoulos, "A fuzzy model for wind speed prediction and power generation in wind parks using spatial correlation," *Energy Conversion, IEEE Transactions on,* vol. 19, pp. 352-361, 2004.

[75] A. U. Haque, P. Mandal, J. Meng, A. K. Srivastava, T. L. Tseng, and T. Senjyu, "A novel hybrid approach based on wavelet transform and fuzzy

ARTMAP networks for predicting wind farm power production," *Industry Applications, IEEE Transactions on,* vol. 49, pp. 2253-2261, 2013.

[76] J. Shi, Z. Ding, W. J. Lee, Y. Yang, Y. Liu, and M. Zhang, "Hybrid forecasting model for very-short term wind power forecasting based on grey relational analysis and wind speed distribution features," *Smart Grid, IEEE Transactions on,* vol. 5, pp. 521-526, 2014.

[77] N. Chen, Q. Zheng, I. T. Nabney, and X. Meng, "Wind power forecasts using Gaussian processes and numerical weather prediction," *Power Systems, IEEE Transactions on,* vol. 29, pp. 656-665, 2014.

[78] J. P. S. Catalao, H. M. I. Pousinho, and V. M. F. Mendes, "Hybrid wavelet-PSO-ANFIS approach for short-term wind power forecasting in Portugal," *Sustainable Energy, IEEE Transactions on,* vol. 2, pp. 50-59, 2011.

[79] Z. H. Zhou, J. Wu, and W. Tang, "Ensembling neural networks: many could be better than all," *Artificial Intelligence,* vol. 137, pp. 239-263, 2002.

[80] G. Brown, J. L. Wyatt, and P. Ti, "Managing diversity in regression ensembles," *Journal of Machine Learning Research,* vol. 6, pp. 1621-1650, 2005.

[81] P. Pinson, H. A. Nielsen, H. Madsen, and G. Kariniotakis, "Skill forecasting from ensemble predictions of wind power," *Applied Energy,* vol. 86, pp. 1326-1334, 2009.

[82] G. Giebel, L. Landberg, J. Badger, K. Sattler, H. Feddersen, and T. S. Nielsen, "Using ensemble forecasting for wind power," in *Proceedings of the 2003 European Wind Energy Conference And Exhibition*, 2003.

[83] S. Lang, C. Mohrlen, J. Jorgensen, B. O. Gallachoir, and E. McKeogh, "Application of a multi-scheme ensemble prediction system for wind power forecasting in Ireland and comparison with validation results from Denmark and Germany," in *Proceedings of the 2006 European Wind Energy Conference and Exhibition*, 2006.

[84] J. W. Taylor, P. E. McSharry, and R. Buizza, "Wind power density

forecasting using ensemble predictions and time series models," *Energy Conversion, IEEE Transactions on,* vol. 24, pp. 775-782, 2009.

[85] L. Duehee and R. Baldick, "Short-term wind power ensemble prediction based on Gaussian processes and neural networks," *Smart Grid, IEEE Transactions on,* vol. 5, pp. 501-510, 2014.

[86] T. Gneiting and M. Katzfuss, "Probabilistic forecasting," *Annual Review of Statistics and Its Application,* vol. 1, pp. 125-151, 2014.

[87] Y. Zhang, J. Wang, and X. Wang, "Review on probabilistic forecasting of wind power generation," *Renewable and Sustainable Energy Reviews,* vol. 32, pp. 255-270, 2014.

[88] B. Abramson and R. Clemen, "Probability forecasting," *International Journal of Forecasting,* vol. 11, pp. 1-4, 1995.

[89] A. U. Haque, M. H. Nehrir, and P. Mandal, "A hybrid intelligent model for deterministic and quantile regression approach for probabilistic wind power forecasting," *Power Systems, IEEE Transactions on,* vol. 29, pp. 1663-1672, 2014.

[90] C. Wan, Z. Xu, P. Pinson, Z. Y. Dong, and K. P. Wong, "Probabilistic forecasting of wind power generation using extreme learning machine," *Power Systems, IEEE Transactions on,* vol. 29, pp. 1033-1044, 2014.

[91] C. Wan, Z. Xu, P. Pinson, Z. Y. Dong, and K. P. Wong, "Optimal prediction intervals of wind power generation," *Power Systems, IEEE Transactions on,* vol. 29, pp. 1166-1174, 2014.

[92] G. B. Huang, Q. Y. Zhu, and C. K. Siew, "Extreme learning machine: a new learning scheme of feedforward neural networks," in *Proceedings of the 2004 IEEE International Joint Conference on Neural Networks*, 2004, pp. 985-990.

[93] X. Chen, Z. Y. Dong, K. Meng, Y. Xu, K. P. Wong, and H. W. Ngan, "Electricity price forecasting with extreme learning machine and bootstrapping," *Power Systems, IEEE Transactions on,* vol. 27, pp. 2055-2062, 2012.

[94] A. H. Nizar, Z. Y. Dong, and Y. Wang, "Power utility nontechnical loss analysis with extreme learning machine method," *Power Systems, IEEE*

Transactions on, vol. 23, pp. 946-955, 2008.

[95] R. Zhang, Z. Y. Dong, Y. Xu, K. Meng, and K. P. Wong, "Short-term load forecasting of Australian national electricity market by an ensemble model of extreme learning machine," *Generation, Transmission & Distribution, IET,* vol. 7, pp. 391-397, 2013.

[96] S. Li, P. Wang, and L. Goel, "Short-term load forecasting by wavelet transform and evolutionary extreme learning machine," *Electric Power Systems Research,* vol. 122, pp. 96-103, 2015.

[97] Y. Xu, Z. Y. Dong, J. H. Zhao, P. Zhang, and K. P. Wong, "A reliable intelligent system for real-time dynamic security assessment of power systems," *Power Systems, IEEE Transactions on,* vol. 27, pp. 1253-1263, 2012.

[98] Y. Xu, Z. Y. Dong, K. Meng, R. Zhang, and K. P. Wong, "Real-time transient stability assessment model using extreme learning machine," *Generation, Transmission & Distribution, IET,* vol. 5, pp. 314-322, 2011.

[99] X. Yan, Z. Y. Dong, Z. Xu, K. Meng, and K. P. Wong, "An intelligent dynamic security assessment framework for power systems with wind power," *Industrial Informatics, IEEE Transactions on,* vol. 8, pp. 995-1003, 2012.

[100] Z. A. Bashir and M. E. El-Hawary, "Applying wavelets to short-term load forecasting using PSO-based neural networks," *Power Systems, IEEE Transactions on,* vol. 24, pp. 20-27, 2009.

[101] M. Hanmandlu and B. K. Chauhan, "Load forecasting using hybrid models," *Power Systems, IEEE Transactions on,* vol. 26, pp. 20-29, 2011.

[102] C. M. Huang and H. T. Yang, "Evolving wavelet-based networks for short-term load forecasting," *Generation, Transmission and Distribution, IEE Proceedings,* vol. 148, pp. 222-228, 2001.

[103] A. S. Pandey, D. Singh, and S. K. Sinha, "Intelligent hybrid wavelet models for short-term load forecasting," *Power Systems, IEEE Transactions on,* vol. 25, pp. 1266-1273, 2010.

[104] C. I. Kim, I. K. Yu, and Y. H. Song, "Kohonen neural network and wavelet transform based approach to short-term load forecasting," *Electric Power Systems Research,* vol. 63, pp. 169-176, 2002.

[105] Z. Bashir and M. E. El-Hawary, "Short term load forecasting by using wavelet neural networks," in *Proceedings of the 2000 Canadian Conference on Electrical and Computer Engineering*, 2000, pp. 163-166.

[106] N. Tai, J. Stenzel, and H. Wu, "Techniques of applying wavelet transform into combined model for short-term load forecasting," *Electric Power Systems Research,* vol. 76, pp. 525-533, 2006.

[107] A. J. R. Reis and A. P. A. da Silva, "Feature extraction via multiresolution analysis for short-term load forecasting," *Power Systems, IEEE Transactions on,* vol. 20, pp. 189-198, 2005.

[108] Q. Y. Zhu, A. K. Qin, P. N. Suganthan, and G. B. Huang, "Evolutionary extreme learning machine," *Pattern Recognition,* vol. 38, pp. 1759-1763, 2005.

[109] D. Karaboga and B. Basturk, "A powerful and efficient algorithm for numerical function optimization: artificial bee colony (ABC) algorithm," *Journal of Global Optimization,* vol. 39, pp. 459-471, 2007.

[110] N. Amjady and F. Keynia, "Short-term load forecasting of power systems by combination of wavelet transform and neuro-evolutionary algorithm," *Energy,* vol. 34, pp. 46-57, 2009.

[111] I. Daubechies, *Ten Lectures on Wavelets*. Philadelphia: Society for Industrial and Applied Mathematics, 1992.

[112] B. L. Zhang and Z. Y. Dong, "An adaptive neural-wavelet model for short term load forecasting," *Electric Power Systems Research,* vol. 59, pp. 121-129, 2001.

[113] N. Saito and G. Beylkin, "Multiresolution representations using the autocorrelation functions of compactly supported wavelets," *Signal Processing, IEEE Transactions on,* vol. 41, pp. 3584-3590, 1993.

[114] R. S. Pathak, *The Wavelet Transform*. Amsterdam: Atlantis Press, 2009.

[115] S. G. Mallat, "A theory for multiresolution signal decomposition: the

wavelet representation," *Pattern Analysis and Machine Intelligence, IEEE Transactions on,* vol. 11, pp. 674-693, 1989.

[116] D. Karaboga and B. Akay, "A comparative study of artificial bee colony algorithm," *Applied Mathematics and Computation,* vol. 214, pp. 108-132, 2009.

[117] D. Karaboga and B. Basturk, "On the performance of artificial bee colony (ABC) algorithm," *Applied Soft Computing,* vol. 8, pp. 687-697, 2008.

[118] D. Karaboga, B. Gorkemli, C. Ozturk, and N. Karaboga, "A comprehensive survey: artificial bee colony (ABC) algorithm and applications," *Artificial Intelligence Review,* pp. 1-37, 2012.

[119] G. Zhu and S. Kwong, "Gbest-guided artificial bee colony algorithm for numerical function optimization," *Applied Mathematics and Computation,* vol. 217, pp. 3166-3173, 2010.

[120] G. B. Huang, Q. Y. Zhu, and C. K. Siew, "Extreme learning machine: theory and applications," *Neurocomputing,* vol. 70, pp. 489-501, 2006.

[121] G. B. Huang, H. Zhou, X. Ding, and R. Zhang, "Extreme learning machine for regression and multiclass classification," *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on,* vol. 42, pp. 513-529, 2012.

[122] P. L. Bartlett, "The sample complexity of pattern classification with neural networks: the size of the weights is more important than the size of the network," *Information Theory, IEEE Transactions on,* vol. 44, pp. 525-536, 1998.

[123] G. B. Huang, D. H. Wang, and Y. Lan, "Extreme learning machines: a survey," *International Journal of Machine Learning and Cybernetics,* vol. 2, pp. 107-122, 2011.

[124] S. McLoone, M. D. Brown, G. Irwin, and G. Lightbody, "A hybrid linear/nonlinear training algorithm for feedforward neural networks," *Neural Networks, IEEE Transactions on,* vol. 9, pp. 669-684, 1998.

[125] I. Guyon and A. Elisseeff, "An introduction to variable and feature selection," *Journal of Machine Learning Research,* vol. 3, pp. 1157-

1182, 2003.

[126] *North American Electric Utility Data*. Available: https://sites.google.com/site/fkeynia/loaddata

[127] P. Shamsollahi, K. W. Cheung, Q. Chen, and E. H. Germain, "A neural network based very short term load forecaster for the interim ISO New England electricity market system," in *Proceedings of the 2001 Power Industry Computer Applications*, 2001, pp. 217-222.

[128] C. Guan, P. B. Luh, L. D. Michel, Y. Wang, and P. B. Friedland, "Very short-term load forecasting: wavelet neural networks with data pre-filtering," *Power Systems, IEEE Transactions on,* vol. 28, pp. 30-41, 2013.

[129] Y. Chen, P. B. Luh, C. Guan, Y. Zhao, L. D. Michel, M. A. Coolbeth*, et al.*, "Short-term load forecasting: similar day-based wavelet neural networks," *Power Systems, IEEE Transactions on,* vol. 25, pp. 322-330, 2010.

[130] A. Deihimi and H. Showkati, "Application of echo state networks in short-term electric load forecasting," *Energy,* vol. 39, pp. 327-340, 2012.

[131] E. Ceperic, V. Ceperic, and A. Baric, "A strategy for short-term load forecasting by support vector regression machines," *Power Systems, IEEE Transactions on,* vol. 28, pp. 4356-4364, 2013.

[132] D. W. Bunn and E. D. Farmer, *Comparative Models for Electrical Load Forecasting*. Chichester, NY: Wiley, 1985.

[133] M. Adya and F. Collopy, "How effective are neural networks at forecasting and prediction? A review and evaluation," *Journal of Forecasting,* vol. 17, pp. 481-495, 1998.

[134] T. M. Peng, N. F. Hubele, and G. G. Karady, "An adaptive neural network approach to one-week ahead load forecasting," *Power Systems, IEEE Transactions on,* vol. 8, pp. 1195-1203, 1993.

[135] D. K. Ranaweera, N. F. Hubele, and A. D. Papalexopoulos, "Application of radial basis function neural network model for short-term load forecasting," *Generation, Transmission and Distribution, IEE Proceedings,* vol. 142, pp. 45-50, 1995.

[136] M. López, S. Valero, C. Senabre, J. Aparicio, and A. Gabaldon, "Application of SOM neural networks to short-term load forecasting: the Spanish electricity market case study," *Electric Power Systems Research,* vol. 91, pp. 18-27, 2012.

[137] A. P. A. da Silva and L. S. Moulin, "Confidence intervals for neural network based short-term load forecasting," *Power Systems, IEEE Transactions on,* vol. 15, pp. 1191-1196, 2000.

[138] Y. Y. Hsu and C. C. Yang, "Design of artificial neural networks for short-term load forecasting. II. multilayer feedforward networks for peak load and valley load forecasting," *Generation, Transmission and Distribution, IEE Proceedings,* vol. 138, pp. 414-418, 1991.

[139] D. C. Park, M. A. El-Sharkawi, R. J. Marks, II, L. E. Atlas, and M. J. Damborg, "Electric load forecasting using an artificial neural network," *Power Systems, IEEE Transactions on,* vol. 6, pp. 442-449, 1991.

[140] P. K. Dash, A. C. Liew, and S. Rahman, "Fuzzy neural network and fuzzy expert system for load forecasting," *Generation, Transmission and Distribution, IEE Proceedings,* vol. 143, pp. 106-114, 1996.

[141] F. J. Marin, F. Garcia-Lagos, G. Joya, and F. Sandoval, "Global model for short-term load forecasting using artificial neural networks," *Generation, Transmission and Distribution, IEE Proceedings,* vol. 149, pp. 121-125, 2002.

[142] J. W. Taylor and R. Buizza, "Neural network load forecasting with weather ensemble predictions," *Power Systems, IEEE Transactions on,* vol. 17, pp. 626-632, 2002.

[143] R. E. Abdel-Aal, "Improving electric load forecasts using network committees," *Electric Power Systems Research,* vol. 74, pp. 83-94, 2005.

[144] M. De Felice and Y. Xin, "Short-term load forecasting with neural network ensembles: a comparative study," *Computational Intelligence Magazine, IEEE,* vol. 6, pp. 47-56, 2011.

[145] M. T. Hagan and M. B. Menhaj, "Training feedforward networks with the Marquardt algorithm," *Neural Networks, IEEE Transactions on,* vol.

5, pp. 989-993, 1994.

[146] S. de Jong, "SIMPLS: an alternative approach to partial least squares regression," *Chemometrics and Intelligent Laboratory Systems,* vol. 18, pp. 251-263, 1993.

[147] P. Geladi and B. R. Kowalski, "Partial least-squares regression: a tutorial," *Analytica Chimica Acta,* vol. 185, pp. 1-17, 1986.

[148] *Partial Least Squares Regression in Matlab*. Available: http://www.mathworks.com/help/stats/plsregress.html

[149] T. M. Cover and J. A. Thomas, *Elements of Information Theory*, 2nd ed. Hoboken: Wiley Interscience, 2006.

[150] L. Yu and H. Liu, "Efficient feature selection via analysis of relevance and redundancy," *Journal of Machine Learning Research,* vol. 5, pp. 1205-1224, 2004.

[151] F. Fleuret, "Fast binary feature selection with conditional mutual information," *Journal of Machine Learning Research,* vol. 5, pp. 1531-1555, 2004.

[152] N. Kwak and C. H. Choi, "Input feature selection by mutual information based on Parzen window," *Pattern Analysis and Machine Intelligence, IEEE Transactions on,* vol. 24, pp. 1667-1671, 2002.

[153] R. S. Marko and K. Igor, "Theoretical and empirical analysis of ReliefF and RReliefF," *Machine Learning,* vol. 53, pp. 23-69, 2003.

[154] L. Breiman, "Random forests," *Machine Learning,* vol. 45, pp. 5-32, 2001.

[155] I. Moghram and S. Rahman, "Analysis and evaluation of five short-term load forecasting techniques," *Power Systems, IEEE Transactions on,* vol. 4, pp. 1484-1491, 1989.

[156] R. E. Abdel-Aal, "Short-term hourly load forecasting using abductive networks," *Power Systems, IEEE Transactions on,* vol. 19, pp. 164-173, 2004.

[157] D. Nguyen and B. Widrow, "Improving the learning speed of 2-layer neural networks by choosing initial values of the adaptive weights," in *Proceedings of the 1990 International Joint Conference on Neural*

Networks, 1990, pp. 21-26.

[158] H. Madsen, P. Pinson, G. Kariniotakis, H. A. Nielsen, and T. Nielsen, "Standardizing the performance evaluation of shor-term wind power prediction models," *Wind Engineering,* vol. 29, pp. 475-489, 2005.

[159] H. Peng, L. Fulmi, and C. Ding, "Feature selection based on mutual information criteria of max-dependency, max-relevance, and min-redundancy," *Pattern Analysis and Machine Intelligence, IEEE Transactions on,* vol. 27, pp. 1226-1238, 2005.

Appendix

A.1 Derivation of Ensemble Output

In this section, the procedure of deriving ensemble output is presented. The data used in Case 6 of Chapter 4 is studied. As mentioned in Section 4.2.2, the proposed ensemble model consists of 24 individual forecasters. Each individual forecaster uses a specific set of wavelet parameters. In the training process, we have 1 target training output (denoted by **PX**) containing the actual load values and 24 individual training outputs (denoted by **PY**) generated by the individual forecasters. In the testing process, 24 individual forecast outputs (denoted by **M**) are obtained by feeding the testing inputs to forecasters. The 24 individual forecast outputs are shown in Figure A.1.

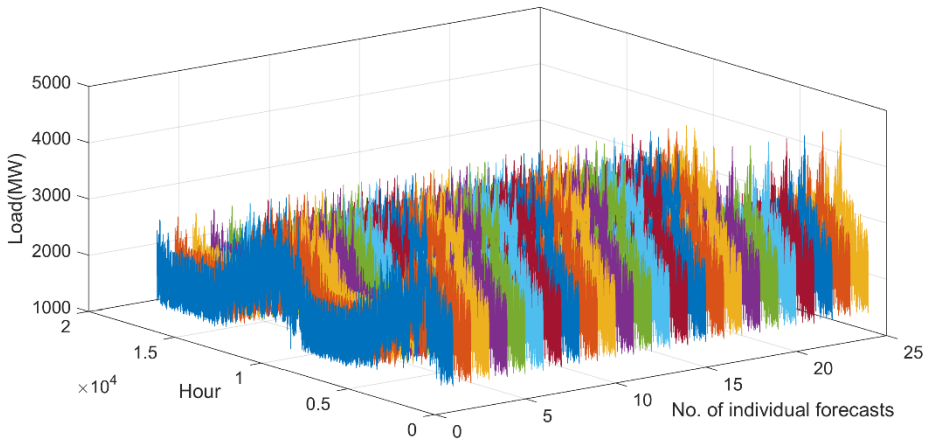


Figure A.1 24 individual forecast outputs.

In the proposed forecasting method, partial least squares regression (PLSR)

is used to generate the ensemble forecast output (denoted by **R**). The ensemble forecast is a weighted average of individual forecasts, where the weight factors are determined by PLSR. Firstly, PLSR is performed between the two variables **PX** and **PY** to obtain the regression coefficients (denoted by **B**). Secondly, with the newly obtained **B**, **R** can be easily predicted from **M**.

In our experiments, the function *plsregress* in Matlab is used to implement PLSR directly [148]. The calculation of outer relations and inner relation can be done accurately and efficiently.

A.2 Selection of Input Variables

In this section, the process of conditional mutual information based feature selection (CMIFS) is clarified. The wind power data used in Case 3 of Chapter 5 is studied. The constitutive component A_2 in two-level decomposition is used. The candidate input feature set of A_2 is $\{A_2(t\text{-}1),\ldots, A_2(t\text{-}100), \mathrm{WS}(t),\ldots, \mathrm{WS}(t\text{-}100)\}$, which includes 201 features. A threshold value T is used to filter out the irrelevant and weakly relevant features, which is set to be 40. The number of selected features in the subset S is 25.

CMIFS can be divided into two phases. In the first phase, the irrelevant or weakly relevant features are removed based on the predefined threshold T. In this phase, the number of features is significantly reduced from 201 to 40. In the second phase, the new feature is selected considering not only the relevance to the dependent variable but also the redundancy to the existing features. In this phase, the number of features is reduced from 40 to 25. The selection results are shown in Table A.1. Note that all the calculations are done in Matlab.

Table A.1 Feature selection results

	Input features	Number of features
Candidate set	$A_2(t\text{-}1),\ldots, A_2(t\text{-}100)$, WS($t$),…, WS($t$-100)	201
1st phase	$A_2(t\text{-}1)$, $A_2(t\text{-}2)$, $A_2(t\text{-}3)$, $A_2(t\text{-}4)$, $A_2(t\text{-}5)$, $A_2(t\text{-}6)$, $A_2(t\text{-}7)$, $A_2(t\text{-}8)$, $A_2(t\text{-}9)$, $A_2(t\text{-}10)$, $A_2(t\text{-}11)$, $A_2(t\text{-}12)$, $A_2(t\text{-}13)$, $A_2(t\text{-}14)$, $A_2(t\text{-}15)$, $A_2(t\text{-}16)$, $A_2(t\text{-}17)$, $A_2(t\text{-}18)$, $A_2(t\text{-}19)$, $A_2(t\text{-}20)$, $A_2(t\text{-}21)$, $A_2(t\text{-}22)$, WS(t-0), WS(t-1), WS(t-2), WS(t-3), WS(t-4), WS(t-5), WS(t-6), WS(t-7), WS(t-8), WS(t-9), WS(t-10), WS(t-11), WS(t-12), WS(t-13), WS(t-14), WS(t-15), WS(t-16), WS(t-17)	40
2nd phase	$A_2(t\text{-}1)$, $A_2(t\text{-}2)$, $A_2(t\text{-}3)$, $A_2(t\text{-}4)$, $A_2(t\text{-}5)$, $A_2(t\text{-}6)$, $A_2(t\text{-}7)$, $A_2(t\text{-}8)$, $A_2(t\text{-}9)$, $A_2(t\text{-}10)$, $A_2(t\text{-}20)$, $A_2(t\text{-}21)$, $A_2(t\text{-}22)$, WS(t-0), WS(t-1), WS(t-2), WS(t-3), WS(t-4), WS(t-5), WS(t-6), WS(t-7), WS(t-8), WS(t-9), WS(t-10), WS(t-11)	25