Electricity Price Forecasting With Extreme Learning Machine and Bootstrapping

Xia Chen, Zhao Yang Dong, *Senior Member, IEEE*, Ke Meng, *Member, IEEE*, Yan Xu, *Student Member, IEEE*, Kit Po Wong, *Fellow, IEEE*, and H. W. Ngan, *Senior Member, IEEE*

***Abstract*—Artificial neural networks (ANNs) have been widely applied in electricity price forecasts due to their nonlinear modeling capabilities. However, it is well known that in general, traditional training methods for ANNs such as back-propagation (BP) approach are normally slow and it could be trapped into local optima. In this paper, a fast electricity market price forecast method is proposed based on a recently emerged learning method for single hidden layer feed-forward neural networks, the extreme learning machine (ELM), to overcome these drawbacks. The new approach also has improved price intervals forecast accuracy by incorporating bootstrapping method for uncertainty estimations. Case studies based on chaos time series and Australian National Electricity Market price series show that the proposed method can effectively capture the nonlinearity from the highly volatile price data series with much less computation time compared with other methods. The results show the great potential of this proposed approach for online accurate price forecasting for the spot market prices analysis.**

***Index Terms*—Bootstrapping, extreme learning machine, interval forecast, price forecast.**

I. Introduction

ELECTRICITY price forecasts is one of the most essential routines for an electricity market. Accurate electricity price points forecasting and prediction intervals are extremely useful for market participants in their risk management and decision making. For suppliers, it is the basic reference to build optimal bidding strategies; for consumers, it helps them derive purchase schedules to obtain maximum power with minimum expenses. Nowadays, smart grid revolution is driving the development of novel price forecasting techniques for electricity market operations and power system analysis. Along with the applications of advanced metering infrastructures (AMI), users' information can be collected and sent to analysis centres, meanwhile, price signal can be sent back to consumers. This interactive two-way communication pattern may influence the manner of electricity consumption heavily, making the system profiles become more fluctuant and unpredictable. Consequently, how to develop an electricity price forecast framework which can provide fast and high-quality forecasting has attracted widespread concerns ever since the establishment of electricity markets.

Many techniques have been employed for this price forecast purpose, including artificial neural networks (ANNs) [1]–[11], fuzzy inference system (FIS) [12], and support vector machines (SVMs) [13], [14]. Moreover, time series models, like ARIMA [15], [16] and GARCH models [17], have also been proven to be effective in the price forecasting/modeling. Among the various electricity price forecasting techniques, especially for the short-term forecasts, ANNs have gained much popularity due to their mature theory background as well as the satisfactory prediction performance. According to [12], the machine learning methods (ANNs, SVMs) outperform other time series methods (ARMA, GARCH). The main advantages of ANNs are their outstanding capabilities in capturing the nonlinear relationships between the input and output datasets. A number of ANNs based prediction models have been proposed and evaluated with real electricity market data. For instance, in [1], a three layer back-propagation neural network (BPNN) was firstly proposed for the Victorian market marginal prices (MMPs) forecasting. In [2] and [3], fuzzy neural networks (FNNs) based predictors were proposed for market clearing prices (MCPs) forecasting for the Ontario and Spanish electricity markets. In [4], a radial basis function neural network (RBFNN) was applied to predict regional MCPs of the Queensland electricity market. In [5] and [6], multiple ANNs were combined together to predict MCPs and confidence intervals in the New England electricity market. Moreover, ANNs could be associated with other mathematic methods to improve forecast accuracy. For instance, in [7], wavelet transforms were used as external decomposer and composer; in [8], mutual information approach was employed for feature selections; similarity methods were applied to select similar days in [9] and [10]; and in [11], a combined ARIMA and ANN model was proposed for MCPs forecast in the Australian national electricity market. Although ANNs based predictors have gained great success in electricity price forecast, it is clear that the traditional training algorithms such as back-propagation method may not be effective enough to train ANNs for predicting price in electricity markets [18]. This shortage has become the main bottleneck blocking their

Manuscript received August 04, 2011; revised January 05, 2012 and March 03, 2012; accepted March 06, 2012. Date of publication April 16, 2012; date of current version October 17, 2012. This work is supported in part by a Hong Kong RGC GRF Grant # 515110 and a Visiting Scholarship of State Key Laboratory of Power Transmission Equipment & System Security and New Technology (Chongqing University) (2007DA10512711401). Paper no. TPWRS-00729-2011.

X. Chen is with Ergon Energy, Brisbane QLD, Australia (e-mail: jessie.chen@ergon.com.au).

Z. Y. Dong, K. Meng, and Y. Xu are with the Center of Intelligent Electricity Networks (CIEN), The University of Newcastle, Callaghan, Australia (e-mail: joe.dong@newcastle.edu.au; kemeng@ieee.org; daniel.xu@uon.edu.au).

K. P. Wong is with the School of Electrical and Electronics Engineering, the University of Western Australia, Perth, Australia (e-mail: kitpo@ieee.org).

H. W. Ngan is with the Department of Electrical Engineering, The Hong Kong Polytechnic University, Hung Hom, Kowloon, Hong Kong (e-mail: eehwngan@polyu.edu.hk).

The authors are also associated with the State Key Laboratory of Power Transmission Equipment & System Security and New Technology (Chongqing University).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TPWRS.2012.2190627

further applications in the price forecasting tasks [19]. The two reasons behind are 1) gradient descent based learning method and 2) iterative tuning of all parameters.

Recently, a novel learning algorithm for single hidden layer feed-forward neural networks (SLFNs) was proposed, called extreme learning machine (ELM) [20], which can overcome the problems caused by gradient descent based approaches. ELM randomly generates all the input weights and the parameters of hidden layer nodes, and then analytically determines the output weights, which means that all the parameters of ELM can be randomly generated or analytically determined instead of being tuned iteratively. ELM and its variants have been verified with a lot of benchmark problems and engineering applications in both regression and classification areas, and shown with faster learning speed and better generalization capability [21]–[26]. In this paper, an ELM based forecasting model is proposed and it is applied to the MCPs forecast. To quantify the uncertainties of ELM based predictions, the bootstrap method based prediction intervals construction framework is adopted as well.

Moreover, estimating the predictive uncertainties associated with points prediction has becoming increasingly important for the risk management purposes. Consequently, for the practical applications, it is necessary to show not only the points forecast results but also the quantitative estimations on the uncertainties behind the results. Two main existing techniques for estimating the prediction uncertainties are 1) the normal approach which gives asymptotic prediction intervals for nonlinear regression based on local linearization model [27], [28]; 2) the bootstrap approach which is based on an imitation of the probabilistic structure of data-generating process on the basis of information provided by the given set of random observations [29]–[31]. As simulation results indicate that a straight-forward application of the first one works poorly in practice, the bootstrapping method becomes more attractive for practical applications.

The paper is structured as follows: after the introduction, the ELM is introduced for completeness, followed by the bootstrap resampling method for prediction intervals construction. Then numerical case studies are carried out and this method is tested with benchmark Mackey-Glass chaos time series before being applied to forecast regional MCPs from the Australian national electricity market (NEM). Conclusions are discussed in the last section.

II. Points and Intervals Forecast With Extreme Learning Machine and Bootstrapping

A. Extreme Learning Machine (ELM)

ELM is an SLFN, with its input weights and biases randomly generated, and its output weights analytically calculated. The critical idea behind ELM is to transform difficult issues arising from nonlinear optimization, like the optimal determination for input weights, hidden layer biases, output weights, to a simple least square problem of deciding the optimal output weights. It means the users do not have to consider all the input weights and the hidden layer biases as long as the norm of weights is small enough, while the output weights are the only issue has to take care. This idea is completely different from the classic iterative learning techniques. Solid mathematical theory and rigid proof have been provided in [20]. The mechanisms of ELM as used in practice can be described as follows.

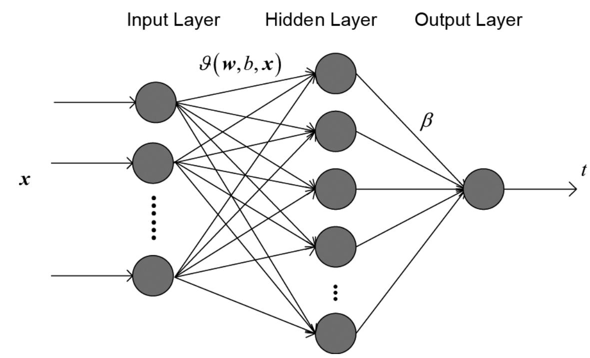

Fig. 1. Typical structure of a MISO ELM.

Suppose an ELM with k hidden layer neurons and activation functions ϑ to model data samples $\{\boldsymbol{x}_i, y_i\}_{i=1}^{n}$, $\boldsymbol{x}_i \in \mathrm{R}^s$, it can be mathematically represented as

$$\sum_{j=1}^{k} \vartheta_j\left(\boldsymbol{w}_j, b_j, \boldsymbol{x}_i\right)\beta_j = t_i, \quad i = 1, 2, \ldots, n \tag{1}$$

where $\boldsymbol{w}_j$ is the input weights vector connecting the jth hidden neuron and inputs, β_j is the output weight connecting the jth hidden neuron and output, and b_j denotes the bias of the jth hidden node. If ELM can approximate the data samples with zero error, we will have $\sum_{i=1}^{n} \|y_i - t_i\| = 0$, i.e., there exist $\boldsymbol{w}_j$, b_j, β_j such that $\sum_{j=1}^{k} \vartheta_j\left(\boldsymbol{w}_j, b_j, \boldsymbol{x}_i\right)\beta_j = y_i$, $i = 1, \ldots, n$. The structure of a multi-input and single-output (MISO) ELM is shown in Fig. 1.

Equation (1) can be written compactly as

$$\boldsymbol{H}\beta = T \tag{2}$$

$$\boldsymbol{H} = \begin{bmatrix} \vartheta\left(\boldsymbol{w}_1, b_1, \boldsymbol{x}_1\right) & & \vartheta\left(\boldsymbol{w}_k, b_k, \boldsymbol{x}_1\right) \\ \vdots & \cdots & \vdots \\ \vartheta\left(\boldsymbol{w}_1, b_1, \boldsymbol{x}_n\right) & & \vartheta\left(\boldsymbol{w}_k, b_k, \boldsymbol{x}_n\right) \end{bmatrix}_{n \times k} \tag{3}$$

where $\boldsymbol{H}$ is the hidden layer output matrix. It has been proven that the input weights vector and hidden bias are not necessarily tuned. Matrix $\boldsymbol{H}$ can actually remain unchanged once random values have been assigned to these parameters in the beginning of learning.

1) If the activation function is infinitely differentiable, when the number of hidden neurons is equal to the number of distinct training samples, $k = n$, the parameters of hidden nodes can be assigned randomly and the output weights can be analytically determined by simply inverting the matrix $\boldsymbol{H}$. Therefore, ELM can approximate data sample with zero error.
2) While for $k \ll n$, $\boldsymbol{H}$ will become a non-square matrix. One specific set of $\tilde{\beta}_j$ could be determined such that

$$Min. \left\|\boldsymbol{H}\left(\boldsymbol{w}_j, b_j, \boldsymbol{x}_i\right)\tilde{\beta}_j - y_i\right\| \tag{4}$$

which is equivalent to minimizing the following cost function:

$$Min.E = \sum_{i=1}^{n}\left[\sum_{j=1}^{k}\vartheta_j\left(\boldsymbol{w}_j, b_j, \boldsymbol{x}_i\right)\tilde{\beta}_j - y_i\right]^2. \quad (5)$$

Therefore, for the given $\boldsymbol{w}_j$ and b_j, (2) becomes a linear system, the output weights can be estimated as

$$\tilde{\beta} = \boldsymbol{H}^{\dagger}T \quad (6)$$

where $\boldsymbol{H}^{\dagger}$ is the Moore-Penrose generalized inverse of matrix $\boldsymbol{H}$. There are several methods to calculate $\boldsymbol{H}^{\dagger}$. It is suggested that singular value decomposition (SVD) is the most appropriate method due to its universality [21].

ELM has been theoretically proven to be capable of universal approximation in a satisfactory sense, and it has been shown to have good generalization capabilities and extremely fast speed. From its algorithm, it can be seen that the only job left for the users is to select activation function and the number of hidden neurons, which make it easy for use. It avoids many difficulties in the conventional learning approaches such as learning rates, learning epochs, stop criteria, and local optima. All these factors motivate us to apply it in electricity MCPs forecasting.

III. Construction of Prediction Intervals by Bootstrapping

The main purpose of this section is to develop an ELM based prediction intervals predictor for MCPs. Basically, there are two sources of uncertainties in ANNs based predictors. One is from the inappropriate selection of ANN architectures. How to select the suitable structures for ANNs is still largely built on a trial-and-error basis. Another one originates from learning approach which may get stuck into local optima, and training process may stop prematurely before reaching global optima [28]. These two uncertainties inherent in predictions given by ANNs means more risks if decision makers take them on faith completely. Consequently, when forecast results are presented to end users, they should be informed of to what extents these results could be trusted. Clearly, the availability of prediction intervals will allow decision makers to efficiently quantify the level of uncertainties associated with the point forecasts, and to consider multiple of solutions for different conditions.

Bootstrap is an approach for estimating the distribution of an estimator or statistic by resampling data [32], [33]. Under mild regularity conditions, the bootstrap yields an approximation to the distribution of an estimator or statistic which is at least as accurate as the approximation results obtained from first-order asymptotic theory. Although the bootstrap approach provides a way to substitute computation for mathematical analysis when it is difficult to obtain asymptotic distribution of an estimator or statistic, the main demerit blocking its practical applications is the high computational cost it involves, which is unacceptable for most of the ANNs. However, since ELM has much faster learning speed when compared to any other learning algorithms, the bootstrap method becomes acceptable.

For given data sample $\{\boldsymbol{x}_i, y_i\}_{i=1}^{n}$, assuming that the residuals $\varepsilon_i = y_i - \vartheta(\boldsymbol{x}_i)$ are independent random variables with finite variance $Var\left(\varepsilon_i^2|\boldsymbol{x}_i\right) = \sigma_\varepsilon^2(\boldsymbol{x}_i)$, function ϑ is approximated by an ELM with k $(k \geq 1)$ hidden layer nodes, then noise variables ε_i can be approximated by

$$\hat{\varepsilon}_i = y_i - \hat{\vartheta}(\boldsymbol{x}_i). \quad (7)$$

For independent estimations, to avoid a systematic error in bootstrapping, $\hat{\varepsilon}_i$ can be centered by

$$\tilde{\varepsilon}_i = \hat{\varepsilon}_i - \frac{1}{n}\sum_{j=1}^{n}\hat{\varepsilon}_j, \quad i = 1, 2, \ldots, n. \quad (8)$$

Let $\tilde{F}$ denote the distribution given by $\tilde{\varepsilon}_1, \tilde{\varepsilon}_2, \ldots, \tilde{\varepsilon}_n$, that is for all i

$$\tilde{\varepsilon}_i^* = \tilde{\varepsilon}_i \text{ with probability } \frac{1}{n}. \quad (9)$$

Create a new bootstrap data sample by adding a randomly sampled residual to the predicted result for each observation to construct new data sample. Therefore, bootstrap residuals can be draw randomly with replacement from $\tilde{\varepsilon}_1, \tilde{\varepsilon}_2, \ldots, \tilde{\varepsilon}_n$ with the probability $1/n$, and then the bootstrap outputs can be formed as

$$y_i^* = \hat{\vartheta}(\boldsymbol{x}_i) + \tilde{\varepsilon}_i^*. \quad (10)$$

Then new bootstrap training data sets can be generated as follows: $\{\boldsymbol{x}_i, y_i^*\}_{i=1}^{n}$. The idea behind bootstrap procedure is that, because ε_i and $\boldsymbol{x}_i$ are independent and identically distributed, for n large enough, sample distributions approximate the true distributions. As a result, ε_i^* and $\boldsymbol{x}_i^*$ behave similar to ε_i and $\boldsymbol{x}_i$. By construction, $\hat{\vartheta}^*$ is close to ϑ, bootstrap outputs show similar random variations as the true outputs.

However, this bootstrap procedure can be applied only to the situations where the noise ε_i does not depend on the input. If $\varepsilon_i = \varepsilon_i(\boldsymbol{x}_i)$ and variance $\sigma_i^2 = Var\left(\varepsilon_i^2|\boldsymbol{x}_i\right)$, wild bootstrap or external bootstrap becomes more appropriate. In the external bootstrap, the independent and identically distributed random variables $\eta_1, \eta_2, \ldots, \eta_n$ with zero mean and unit variance can be generated. With residuals $\hat{\varepsilon}_i = y_i - \hat{\vartheta}(\boldsymbol{x}_i)$, the $\tilde{\varepsilon}_i^*$ is centred fitted residuals. After that, $\tilde{\varepsilon}_i^*$ will be transformed randomly by multiplying it with η_i, and bootstrap outputs can be defined as

$$y_i^* = \hat{\vartheta}(\boldsymbol{x}_i) + \eta_i\tilde{\varepsilon}_i^*. \quad (11)$$

In contrast to the standard bootstrap, the bootstrap noise $\hat{\varepsilon}_i^* = \eta_i\tilde{\varepsilon}_i^*$ is generated in a manner depending on the bootstrap input $\boldsymbol{x}_i$ to reflect the dependence of $\hat{\varepsilon}_i$ on the input in the original training data set.

Once ψ set of bootstrap replicates are obtained, there are several methods to constructing bootstrap prediction intervals, like the normal-theory interval and bootstrap percentile interval [34], [35]. If the number of bootstrap replications is large, the quantiles of the bootstrap sampling distribution of the estimator can be used to establish prediction interval nonparametrically.

The $100(1-\alpha)\%$ prediction intervals are defined as

$$Q_{(lower)} < y_i < Q_{(upper)} \quad (12)$$

where $Q_{(1)}, \ldots, Q_{(\psi)}$ are the ordered bootstrap replicates, and $lower = \psi\alpha/2$, $upper = \psi(1-\alpha/2)$. If lower and upper are not integers, then we can interpolate between adjacent ordered values or round off to the nearest integer. One question that remains unanswered is how to select ψ. The constructions of prediction intervals need accurate information of the lower and upper quantiles of the distribution. Therefore, enough replicates should be generated, in our case study, ψ is chosen as 10 000.

The above steps are summarized as follows for completeness.

ELM-Bootstrap Prediction Intervals

Step-1) Given data $\{(\boldsymbol{x}_1, y_1), \ldots, (\boldsymbol{x}_n, y_n)\}$, calculate $\hat{\vartheta}(\boldsymbol{x}_i)$ using ELMs.

Step-2) Compute residuals $\hat{\varepsilon}_i = y_i - \hat{\vartheta}(\boldsymbol{x}_i)$.

Step-3) Re-center residuals $\tilde{\varepsilon}_i = \hat{\varepsilon}_i - n^{-1}\sum_{j=1}^{n}\hat{\varepsilon}_j$.

Step-4) Generate new bootstrap data $y_i^* = \hat{\vartheta}(\boldsymbol{x}_i) + \eta_i\tilde{\varepsilon}_i^*$.

Step-5) Calculate $\hat{\vartheta}^*(\boldsymbol{x}_i)$ using ELMs from $\{(\boldsymbol{x}_1, y_1^*), \ldots, (\boldsymbol{x}_n, y_n^*)\}$.

Step-6) Repeat **Steps 4–6** to get ψ sets of bootstrap replicates.

Step-7) Order bootstrap replicates, and compute prediction intervals

IV. Chaos Time Series Analysis

In order to mathematically illustrate the effectiveness of the proposed approach in time series analysis, a benchmark chaos time series is used as the first case study.

A. Data Description

In this case study, short-term and long-term Mackey-Glass chaos time series [36] are used, which can be represented as

$$x_{i+1} = a\frac{x_{i-\tau}}{1 + x_{i-\tau}^{c}} + (1-b)\,x_i. \tag{13}$$

When $\tau \geq 17$, the time series produced by (13) represents chaos characteristics. Because of the high complexity, this data series seem to be random series. Accordingly, this equation is usually applied to test the learning and generation capabilities of ANNs. The parameters are set as follows:

$$a = 0.2, \quad b = 0.1, \quad c = 10, \quad \tau = 17. \tag{14}$$

Mackey-Glass time series long-term predictions are more difficult than short-term predictions. It requires more neurons and longer training time. The following function is chosen to approximate the Mackey-Glass chaos time series:

$$\begin{cases} x_{i+\varphi} = \vartheta\,(x_i, \ldots, x_{i-2}, x_{i-3}), & \varphi = 5, 10 \\ x_0, x_1, \ldots, x_{17} = 1.2. \end{cases} \tag{15}$$

Based on moving window, totally 50 independent forecasts are applied to check the availability and generalization ability. To evaluate the performance of the proposed approach, three criteria are adopted, to namely mean absolute error (MAE), mean absolute percentage error (MAPE), and root mean square error (RMSE). In order to demonstrate the predict performance clearly, five machine learning methods BPNN, RBFNN, FNN, SVM, and RVM are used in the case study:

$$\begin{cases} \text{MAE} = \frac{1}{n}\sum\limits_{i=1}^{n}|y_i - t_i| \\ \text{MAPE} = \frac{1}{n}\sum\limits_{i=1}^{n}\frac{|y_i - t_i|}{y_i} \times 100\% \\ \text{RMSE} = \sqrt{\frac{1}{n}\sum\limits_{i=1}^{n}(y_i - t_i)^2}. \end{cases} \tag{16}$$

TABLE I
Summary of Chaos Time Series Forecast Results

Step	Stage	Model	MAE	MAPE	RMSE
N=5	Training	BPNN	0.0142	1.5807%	0.0190
		RBFNN	0.0123	1.3417%	0.0166
		FNN	0.0189	2.1239%	0.0241
		SVM	0.0423	4.6520%	0.0575
		RVM	0.0251	2.9139%	0.0331
		ELM	0.0123	1.3354%	0.0166
	Testing	BPNN	0.0148	1.6364%	0.0190
		RBFNN	0.0146	1.6134%	0.0189
		FNN	0.0191	2.2170%	0.0236
		SVM	0.0474	5.3385%	0.0638
		RVM	0.0352	4.1813%	0.0514
		ELM	**0.0141**	**1.5374%**	**0.0184**
N=10	Training	BPNN	0.0264	2.9285%	0.0351
		RBFNN	0.0222	2.4515%	0.0299
		FNN	0.0357	3.8881%	0.0457
		SVM	0.0575	6.3129%	0.0929
		RVM	0.0422	5.0061%	0.0550
		ELM	0.0195	2.2131%	0.0262
	Testing	BPNN	0.0306	3.3276%	0.0394
		RBFNN	0.0308	3.3615%	0.0393
		FNN	0.0381	4.2011%	0.0480
		SVM	0.0846	9.2764%	0.1257
		RVM	0.0637	7.6286%	0.0927
		ELM	**0.0287**	**3.1276%**	**0.0371**

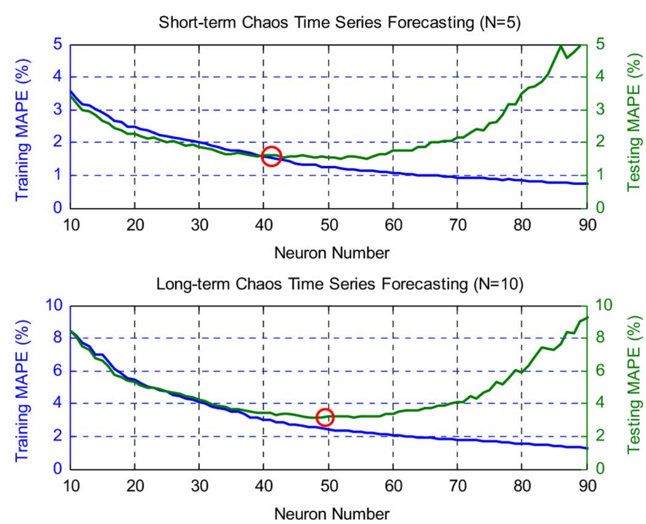

Fig. 2. Comparisons of different hidden layer neurons for ELM.

B. Forecasting Performance

Based on the data obtained through 50 independent forecasts, the comparisons of average prediction performance by different approaches are presented in Table I.

The hidden node number of all the ANNs (BPNN, RBFNN, FNN, and ELM) and the parameters for vector machines (SVM and RVM) are fine-tuned using leave-one-out cross validation [37] and differential evolution [38] based on the MAPE value. The impacts of different hidden node number on ELM prediction performance are shown in Fig. 2. We can see that the optimized model can effectively balance the learning and generalization performance.

The average computational costs of 50 independent forecasts for each method are given in Table II. The testing environment is a desktop with 2.66-GHz CPU and 2.0-GB physical memory.

TABLE II
SUMMARY OF COMPUTATION TIME

Model	Computation Time (s)	
	Short-term Forecasting	Long-term Forecasting
BPNN	2.2999	2.8716
RBFNN	0.1750	0.1820
FNN	1.2144	1.2210
SVM	30.9649	31.3872
RVM	0.1948	0.2065
ELM	**0.0419**	**0.0441**

TABLE III
SUMMARY OF MCPs OF QLD MARKET (AUD/MWh)

Season	Mean	Std.	Minimum	Median
Winter	26.50	75.91	9.50	19.00
Spring	22.01	13.93	7.67	18.77
Summer	43.99	229.76	5.28	22.68
Autumn	61.18	56.69	13.07	53.34

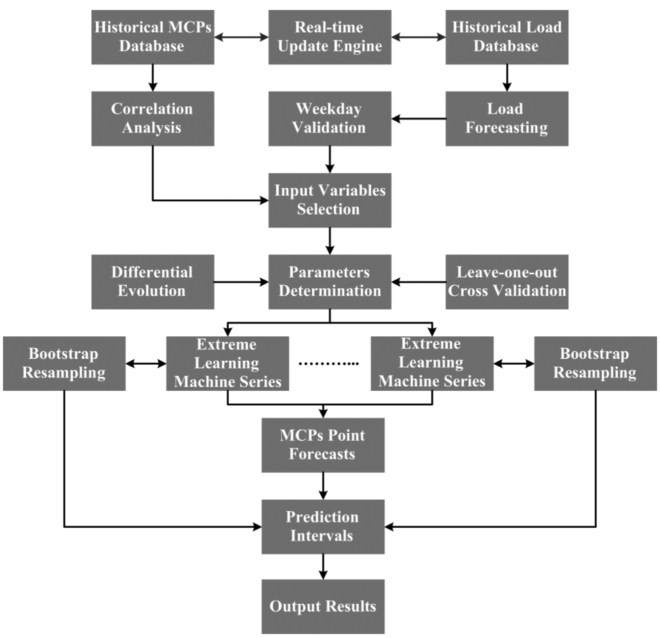

Fig. 3. Framework of ELM-bootstrap for MCPs forecasting.

C. Results Analysis

1) Note that each approach is typically associated with a few model parameters that need to be provided/tuned in advance. We suggest that in order to obtain superior performance, these parameters should have problem-oriented values.
2) It is clear that ELM outperforms the other approaches in both short-term and long-term chaos time series forecasts, with regards to the given criteria.
3) It can be seen clearly that the ELM method require much less time in each case study; that is because of the unique learning mechanism.
4) All these results establish the ground of further application of the proposed approach in electricity MCPs analysis.

V. ELECTRICITY MARKET CLEARING PRICE FORECASTING

The purpose of this case study is to compare the efficiency of the proposed forecast model with other traditional methods.

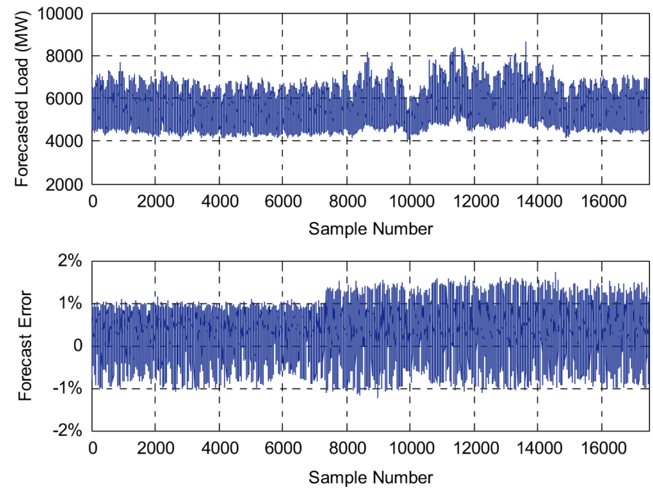

Fig. 4. Load forecasting results of QLD market (June 1, 2006–May 31, 2007).

To achieve this goal, the Australian NEM is selected as a candidate market. Trading in the NEM is based on a 30-min trading interval. Generators submit their offers every 5 min each day. Dispatch price is determined every 5 min, 288 prices in a day, and 6 dispatch prices are averaged every half-hour to determine the regional MCPs which is the basis for settlement of financial transactions for all energy traded in the NEM. In order to assist decision-making process for generators, there are totally 48 MCPs needed to be predicted at the same time for the coming trading day.

A. Data Description

The market price data are taken from the Australian Energy Market Operator (AEMO) website [39]. The dataset used in the study consists of whole one year MCPs from QLD market. Data period is from June 1, 2006 to May 31, 2007, including a total of 17 520 observations. There are four seasons, winter (Jun.–Aug.), spring (Sep.–Nov.), summer (Dec.–Feb.), and autumn (Mar.–May) in Australia. Therefore, four datasets each contain three months' MCPs are constructed, which represent the variations of demand and price of different seasons over a year in the QLD market. A summary of the MCPs based on different seasons is shown in Table III.

The proposed overall forecasting framework is shown in Fig. 3. As each trading interval displays a rather distinct price profiles reflecting the daily variations of system demand and operation constraints, the modeling is implemented separately on every trading interval across one day. Therefore, different model specifications are selected for each individual trading interval based on special price dynamic profiles. Due to unique characteristics, the future MCPs can be estimated based on the historical observations of past days, weeks, or years at the same trading interval, as well as specific fluctuation rules. Therefore, correlation analysis should be applied to investigate the relations between future price trajectory and previous observations at different lag time to determine the potential number of model inputs. Furthermore, in order to recognize the weekly pattern of MCPs, three dummy variables are applied to capture the special time variance on Saturday, Sunday, and workdays. The demand is forecasted by our commercial software OptiLoad [40]. The forecast load profile of the QLD market over one year is shown in Figs. 4 and 5, and is summarized in Table IV.

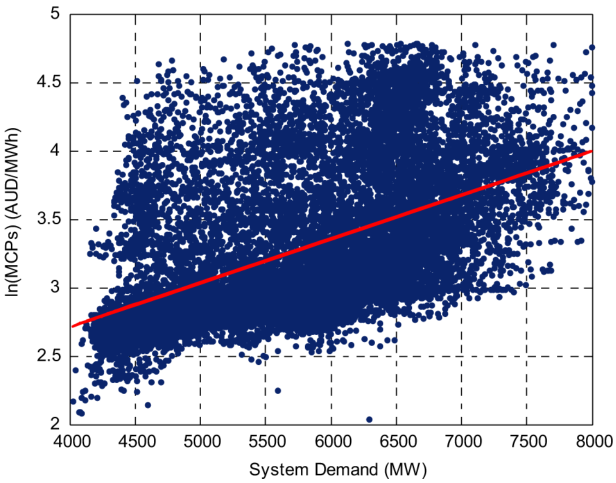

Fig. 5. ln (MCPs) versus system demand.

TABLE IV
SUMMARY OF LOAD FORECASTING ERROR

Season	Min. (%)	Max. (%)	Mean (%)	Std.
Winter	0	1.42	0.31	0.27
Spring	1.40e-4	1.53	0.29	0.27
Summer	0	1.60	0.32	0.30
Autumn	0	1.41	0.32	0.30

B. Points Forecast

In the following case study, seven consecutive days in the last month of each season are selected to test the performance of each model. Corresponding MCPs point and prediction interval forecasting results are provided, respectively.

Based on the performance of numerical case study, another two machine leaning methods BPNN and RBFNN are used as well. To assess the performance of individual model in terms of points forecasting, three criteria are used: MAE, MAPE, and RMSE. The comparisons of 50 independent forecasts for each season by different approaches are presented in Table V.

C. Prediction Intervals

Normally by comparing the nominal coverage of predicted intervals to the true coverage, the effectiveness of this model in terms of interval forecasting can be estimated. We can calculate prediction intervals and determine the actual percentage of the exceedances at different levels.

For given m out-of-sample observations and corresponding forecasted interval $[l_i, u_i]$ under level α, empirical confidence $\hat{\alpha}$ and absolute coverage error (ACE) can be defined as

$$\hat{\alpha} = \frac{\text{Frequency}\,(t_i \in [l_i, u_i])}{m} \tag{17}$$

$$\text{ACE} = |\hat{\alpha} - \alpha| \tag{18}$$

where $\hat{\alpha}$ is the percentage of observations which fall into the forecast intervals. The average results of MCP forecast by ELM in winter is given in Fig 6.

As the results shown in Table VI, the empirical coverage and ACE from nominal coverage for the models based on different testing sets are provided. It can be seen that the derivation of empirical coverage from nominal coverage of interval forecasts by theoretical formulae is bigger than using bootstrap interval construction.

TABLE V
SUMMARY OF MCP FORECAST RESULTS

Season	Stage	Model	MAE	MAPE	RMSE
Winter	Training	BPNN	1.1566	5.1936%	1.5225
		RBFNN	0.8345	3.8910%	1.1430
		ELM	0.9458	4.3466%	1.3544
	Testing	BPNN	2.3611	9.9423%	3.3470
		RBFNN	2.1046	8.5440%	3.0537
		ELM	**2.0278**	**8.3372%**	**2.9371**
Spring	Training	BPNN	1.3195	6.2802%	1.7272
		RBFNN	1.0048	4.7648%	1.3311
		ELM	1.2548	5.9897%	1.6882
	Testing	BPNN	**2.2337**	**9.9291%**	**3.1190**
		RBFNN	2.6382	11.5712%	3.6026
		ELM	2.3021	10.2642%	3.1781
Summer	Training	BPNN	4.4040	15.3203%	6.0542
		RBFNN	3.1721	11.4565%	4.3307
		ELM	3.7803	12.3331%	5.5882
	Testing	BPNN	10.9983	24.4636%	17.2313
		RBFNN	10.8783	22.7230%	17.7526
		ELM	**10.1656**	**21.8798%**	**16.5881**
Autumn	Training	BPNN	5.3081	9.6121%	6.7923
		RBFNN	3.9860	7.6349%	5.1718
		ELM	5.5116	10.0494%	6.8658
	Testing	BPNN	7.8198	13.7900%	10.7256
		RBFNN	7.9618	13.5447%	11.4401
		ELM	**7.3193**	**12.7363%**	**10.3818**

D. Comparisons of Yearly Forecast Results

The yearly RMSE and MAPE of ELM are compared to those of ARMA, GARCH, BPNN, RBFNN, FNN, SVM, and RVM, in Fig. 7 for completeness. The observation is that ARMA and GARCH are less accurate than all others in terms of two criteria. FNN, SVM, and BPNN have similar levels of accuracy, while ELM, RVM, and RBFNN have lower predict errors. Regarding to the computational cost, ELM, RVM, RBFNN, and BPNN are much faster than ARMA, GARCH, SVM, and FNN.

E. Results Analysis

1) Majority of forecast approaches can provide predictions of electricity price series in a future time; however, there is no 100% reliable forecast. As is clearly shown in results, generally ELM outperforms other methods in most of the testing datasets, with regards to the given criteria.
2) ELM saves lots of computation time and system memory, and requires less user-defined model parameters, which is suitable for large-scale data analysis in practical applications. Owing to its high efficient tuning mechanism, ELM model can be online updated to maintain performance in case of significant pattern changes in data series.
3) We should state that it is unpractical to predict all the data with single model because it is a nontrivial task and handling process would be very complicated due to large amount of data involved, and sometimes results are not good.
4) We believe that the optimal model is a composite forecast model which can statistically produce an optimal forecast by computing prediction results from a number of different methods, taking into account historical observations and other relevant information such as demand, maintenance schedules, weather conditions, and so on. Furthermore, in this composite model, results from different prediction models should be processed by adaptive processing module which adjusts the weights of each model based on

TABLE VI
SUMMARY OF INTERVAL FORECASTING RESULTS

Season	α	Model	$\hat{\alpha}$	ACE
		BPNN	85.42%	5.42%
	80.00%	RBFNN	83.63%	3.63%
		ELM	**80.95%**	**0.95%**
		BPNN	90.48%	0.48%
Winter	90.00%	RBFNN	88.69%	1.31%
		ELM	**89.88%**	**0.12%**
		BPNN	95.83%	3.17%
	99.00%	RBFNN	97.02%	1.98%
		ELM	**97.92%**	**1.08%**
		BPNN	85.12%	5.12%
	80.00%	RBFNN	83.63%	3.63%
		ELM	**82.74%**	**2.74%**
		BPNN	**91.67%**	**1.67%**
Spring	90.00%	RBFNN	93.15%	3.15%
		ELM	92.26%	2.26%
		BPNN	**97.32%**	**1.68%**
	99.00%	RBFNN	97.02%	1.98%
		ELM	97.32%	1.68%
		BPNN	86.90%	6.90%
	80.00%	RBFNN	87.50%	7.50%
		ELM	**85.42%**	**5.42%**
		BPNN	91.07%	1.07%
Summer	90.00%	RBFNN	89.58%	0.42%
		ELM	**89.88%**	**0.12%**
		BPNN	97.32%	1.68%
	99.00%	RBFNN	**97.32%**	**1.68%**
		ELM	97.62%	1.38%
		BPNN	87.20%	7.20%
	80.00%	RBFNN	85.71%	5.71%
		ELM	**82.74%**	**2.74%**
		BPNN	**90.18%**	**0.18%**
Autumn	90.00%	RBFNN	91.67%	1.67%
		ELM	89.29%	0.71%
		BPNN	96.73%	2.27%
	99.00%	RBFNN	96.73%	2.27%
		ELM	**98.21%**	**0.79%**

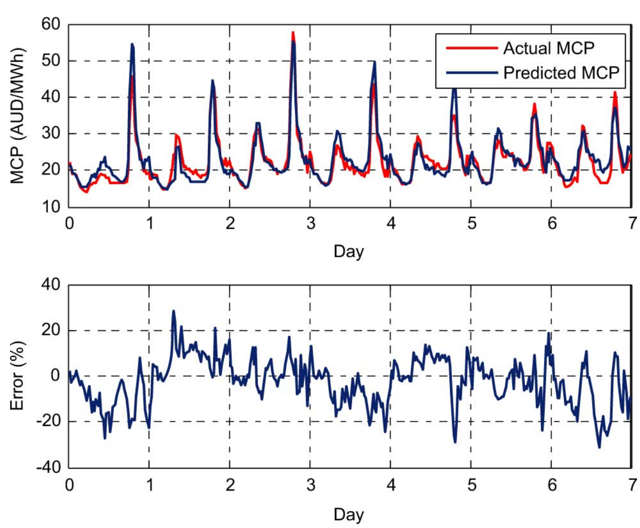

Fig. 6. Average results of MCP forecast by ELM in winter.

their prediction performance on a continuous basis. Meanwhile, human expert input to select the preferred prediction results also need to be incorporated to best enhance the performance of forecast tool. We will report the relevant research results in our successive publications.

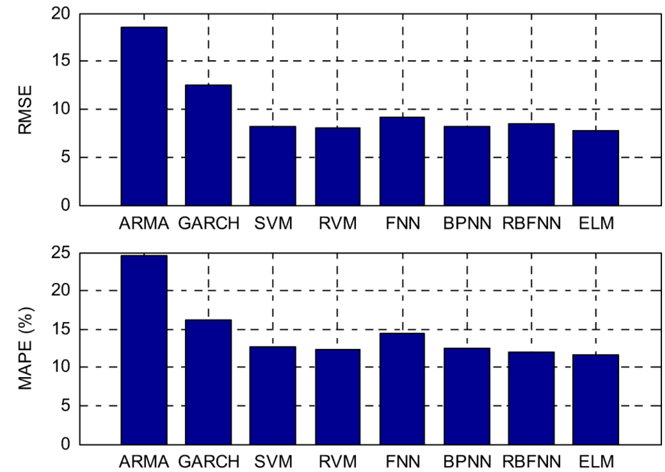

Fig. 7. Yearly RMSE and MAPE of forecasted QLD MCPs.

VI. CONCLUSION

In this paper, an ELM-Bootstrap electricity MCPs forecasting framework is proposed. In this model, ELM is firstly used to predict price points. To quantify the uncertainties of prediction results, bootstrap based prediction intervals construction is applied. The availability of prediction intervals allows decision maker to efficiently quantify the level of uncertainty associated with point forecasts, which best suits the needs for electricity market operations and risk management purposes. Using chaos time series and electricity market data samples from Australian NEM, the case study results illustrate that the ELM-Bootstrap MCPs forecasting framework can be used as an accessorial tool by electricity market participants.

REFERENCES

[1] B. R. Szkuta, L. A. Sanabria, and T. S. Dillon, "Electricity price short-term forecasting using artificial neural networks," *IEEE Trans. Power Syst.*, vol. 14, no. 3, pp. 851–857, Aug. 1999.

[2] C. P. Rodriguez and G. J. Anders, "Energy price forecasting in the Ontario competitive power system market," *IEEE Trans. Power Syst.*, vol. 19, no. 1, pp. 366–374, Feb. 2004.

[3] N. Amjady, "Day-ahead price forecasting of electricity markets by a new fuzzy neural network," *IEEE Trans. Power Syst.*, vol. 21, no. 2, pp. 887–896, May 2006.

[4] K. Meng, Z. Y. Dong, and K. P. Wong, "Self-adaptive RBF neural network for short-term electricity price forecasting," *IET Gen., Transm., Distrib.*, vol. 3, no. 4, pp. 325–335, Apr. 2009.

[5] L. Zhang, P. B. Luh, and K. Kasiviswanathan, "Energy clearing price prediction and confidence interval estimation with cascaded neural networks," *IEEE Trans. Power Syst.*, vol. 18, no. 1, pp. 99–105, Feb. 2003.

[6] J. J. Guo and P. B. Luh, "Improving market clearing price prediction by using a committee machine of neural networks," *IEEE Trans. Power Syst.*, vol. 19, no. 4, pp. 1867–1876, Nov. 2004.

[7] N. M. Pindoriya, S. N. Singh, and S. K. Singh, "An adaptive wavelet neural network-based energy price forecasting in electricity markets," *IEEE Trans. Power Syst.*, vol. 23, no. 3, pp. 1423–1432, Aug. 2008.

[8] N. Amjady and F. Keynia, "Day-ahead price forecasting of electricity markets by mutual information techniques and cascaded neuroevolutionary algorithm," *IEEE Trans. Power Syst.*, vol. 24, no. 1, pp. 306–318, Feb. 2009.

[9] P. Mandal, T. Senjyu, N. Urasaki, T. Funabashi, and A. K. Srivastava, "A novel approach to forecast electricity price for pjm using neural network and similar days method," *IEEE Trans. Power Syst.*, vol. 22, no. 4, pp. 2058–2065, Nov. 2007.

[10] P. Mandal, A. K. Srivastava, and J. W. Park, "An effort to optimize similar days parameters for ANN-based electricity price forecasting," *IEEE Trans. Ind. Appl.*, vol. 45, no. 5, pp. 1888–1896, Sep./Oct. 2009.

[11] P. Areekul, T. Senjyu, H. Toyama, and A. Yona, "A hybrid ARIMA and neural network model for short-term price forecasting in deregulated market," *IEEE Trans. Power Syst.*, vol. 25, no. 1, pp. 524–530, Feb. 2010.

[12] G. Li, C. C. Liu, C. Mattson, and J. Lawarree, "Day-ahead electricity price forecasting in a grid environment," *IEEE Trans. Power Syst.*, vol. 22, no. 1, pp. 266–274, Feb. 2007.
[13] J. H. Zhao, Z. Y. Dong, X. Li, and K. P. Wong, "A framework for electricity price spike analysis with advanced data mining methods," *IEEE Trans. Power Syst.*, vol. 22, no. 1, pp. 376–385, Feb. 2007.
[14] J. H. Zhao, Z. Y. Dong, Z. Xu, and K. P. Wong, "A statistical approach for interval forecasting of the electricity price," *IEEE Trans. Power Syst.*, vol. 23, no. 2, pp. 267–276, May 2008.
[15] J. Contreras, R. Espinola, F. J. Nogales, and A. J. Conejo, "Arima models to predict next-day electricity prices," *IEEE Trans. Power Syst.*, vol. 18, no. 3, pp. 1014–1020, Aug. 2003.
[16] A. J. Conejo, M. A. Plazas, R. Espinola, and A. B. Molina, "Day-ahead electricity price forecasting using the wavelet transform and ARIMA models," *IEEE Trans. Power Syst.*, vol. 20, no. 2, pp. 1035–1042, May 2005.
[17] R. C. Garcia, J. Contreras, M. V. Akkeren, and J. B. C. Garcia, "A GARCH forecasting model to predict day-ahead electricity prices," *IEEE Trans. Power Syst.*, vol. 20, no. 2, pp. 867–874, May 2005.
[18] A. M. Gonzalez, A. M. S. Roque, and J. G. Gonzalez, "Modeling and forecasting electricity prices with input/output hidden Markov models," *IEEE Trans. Power Syst.*, vol. 20, no. 1, pp. 13–24, Feb. 2005.
[19] K. Meng, Z. Y. Dong, X. Yin, and K. P. Wong, "Electricity market clearing price forecasting," in *Computational Intelligence in Power Systems*, L. B. Shi and Z. Dong, Eds. New York: Springer, 2009, pp. 89–106.
[20] G. B. Huang, Q. Y. Zhu, and C. K. Siew, "Extreme learning machine: theory and applications," *Neurocomputing*, vol. 70, no. 1–3, pp. 489–501, Dec. 2006.
[21] G. B. Huang, L. Chen, and C. K. Siew, "Universal approximation using incremental constructive feedforward networks with random hidden nodes," *IEEE Trans. Neural Netw.*, vol. 17, no. 4, pp. 879–892, Jul. 2006.
[22] G. B. Huang and L. Chen, "Convex incremental extreme learning machine," *Neurocomputing*, vol. 70, no. 16–18, pp. 3056–3062, Oct. 2007.
[23] G. B. Huang and L. Chen, "Enhanced random search based incremental extreme learning machine," *Neurocomputing*, vol. 71, no. 16–18, pp. 3460–3468, Oct. 2008.
[24] G. Feng, G. B. Huang, Q. Lin, and R. Gay, "Error minimized extreme learning machine with growth of hidden nodes and incremental learning," *IEEE Trans. Neural Netw.*, vol. 20, no. 8, pp. 1352–1357, Aug. 2009.
[25] A. H. Nizar, Z. Y. Dong, and Y. Wang, "Power utility nontechnical loss analysis with extreme learning machine method," *IEEE Trans. Power Syst.*, vol. 23, no. 3, pp. 946–955, Aug. 2008.
[26] Y. Xu, Z. Y. Dong, K. Meng, R. Zhang, and K. P. Wong, "A new real-time transient stability assessment model using extreme learning machine," *IET Gen., Transm., Distrib.*, vol. 5, no. 3, pp. 314–322, Mar. 2011.
[27] J. T. G. Hwang and A. A. Ding, "Prediction intervals for artificial neural networks," *J. Amer. Statist. Assoc.*, vol. 92, no. 438, pp. 748–757, Jun. 1997.
[28] R. D. De Veaux, J. Schweinsberg, J. Schumi, and L. H. Ungar, "Prediction intervals for neural networks via nonlinear regression," *Technometrics*, vol. 40, no. 4, pp. 273–282, Nov. 1998.
[29] W. G. Baxt and H. White, "Bootstrapping confidence intervals for clinical input variable effects in a network trained to identify the presence of acute myo-cardial infarction," *Neural Comput.*, vol. 7, no. 3, pp. 624–638, May 1995.
[30] J. Franke and M. H. Neumann, "Bootstrapping neural networks," *Neural Comput.*, vol. 12, no. 8, pp. 1929–1949, Aug. 2000.
[31] F. Giordano, M. L. Rocca, and C. Perna, "Forecasting nonlinear time series with neural network sieve bootstrap," *Comput. Statist. Data Anal.*, vol. 51, no. 8, pp. 3871–3884, May 2007.
[32] B. Efron, "Bootstrap methods: another look at the jackknife," *Ann. Statist.*, vol. 7, no. 1, pp. 1–26, Jan. 1979.
[33] B. Efron and R. J. Tibshirani, "An introduction to the bootstrap," in *Monographs on Statistics & Applied Probability*. London, U.K.: Chapman & Hall/CRC, 1993.
[34] D. A. Freedman, "Bootstrapping regression models," *Ann. Statist.*, vol. 9, no. 6, pp. 1218–1228, 1981.
[35] J. Fox, "Bootstrapping regression models," Appendix to *An R and S-PLUS Companion to Applied Regression*, Jan. 2002.
[36] M. C. Mackey and J. Glass, "Oscillation and chaos in physiological control systems," *Science*, vol. 197, no. 4300, pp. 287–289, Jul. 1977.
[37] H. Zhu and R. Rohwer, "No free lunch for cross-validation," *Neural Comput.*, vol. 8, no. 7, pp. 1421–1426, Oct. 1996.
[38] K. Meng, Z. Y. Dong, D. H. Wang, and K. P. Wong, "A self-adaptive RBF neural network classifier for transformer fault analysis," *IEEE Trans. Power Syst.*, vol. 25, no. 3, pp. 1350–1360, Aug. 2010.
[39] Australian Energy Market Operator (AEMO). [Online]. Available: http://www.aemo.com.au.
[40] OptiLoad V1.0 User Manual, PolyU, 2009.

Xia Chen received the Ph.D. degree from the University of Queensland, Brisbane, Australia, in 2010.

She is currently a Market Analyst with Ergon Energy, Brisbane, Australia. Her research interest includes electricity physical and financial market analysis, demand/price forecasting and risk management, etc.

Zhao Yang Dong (M'99–SM'06) received the Ph.D. degree from the University of Sydney, Sydney, Australia, in 1999.

He is now Ausgrid Chair Professor and Director of the Centre for Intelligent Electricity Networks (CIEN), The University of Newcastle, Callaghan, Australia. He previously held academic and industrial positions with the Hong Kong Polytechnic University, the University of Queensland, Australia, and Transend Networks, Tasmania, Australia. His research interest includes smart grid, power system planning, power system security, stability and control, load modeling, electricity market, and computational intelligence and its application in power engineering.

Prof. Dong is an editor of the IEEE TRANSACTIONS ON SMART GRID and an editor of the IEEE POWER ENGINEERING LETTERS.

Ke Meng (M'10) received the Ph.D. degree from The University of Queensland, Brisbane, Australia.

He is with the Centre for Intelligent Electricity Networks, The University of Newcastle, Callaghan, Australia. He was a research fellow at the Department of Electrical Engineering, The Hong Kong Polytechnic University, Hong Kong. His research interest includes pattern recognition, power system stability analysis, wind power, and energy storage.

Yan Xu (S'10) received the B.E. and M.E. degrees from South China University of Technology, Guangzhou, China. He is now pursuing the Ph.D. degree at The University of Newcastle, Callaghan, Australia. His research interests include power system stability and control, power system planning, renewable energy, and intelligent system applications to power engineering.

Kit Po Wong (M'87–SM'90–F'02) received the M.Sc, Ph.D., and D.Eng. degrees from the University of Manchester, Institute of Science and Technology, Manchester, U.K., in 1972, 1974, and 2001, respectively. He is with The University of Western Australia, Perth, Australia, from 1974 until 2004 and from 2011. Since 2002, he had been Chair Professor, and previously Head, of the Department of Electrical Engineering, The Hong Kong Polytechnic University, Kowloon, Hong Kong. His current research interests include computation intelligence applications to power system analysis, planning and operations, as well as power market analysis.

Prof. Wong received three Sir John Madsen Medals (1981, 1982, and 1988) from the Institution of Engineers Australia, the 1999 Outstanding Engineer Award from IEEE Power Chapter Western Australia, and the 2000 IEEE Third Millennium Award. He was a Co-Technical Chairman of the IEEE ICMLC 2004 Conference and General Chairman of IEEE/CSEE PowerCon2000. He was an Editor-in-Chief of *IEE Proceedings in Generation, Transmission and Distribution* and Editor (Electrical) of the *Transactions of Hong Kong Institution of Engineers*. He is the Editor-in-Chief of the IEEE POWER ENGINEERING LETTERS. He is a Fellow of IET, HKIE, and IEAust.

H. W. Ngan (SM'01) received the M.Sc. degree from the University of Aston in 1979, the M.B.A. degree from the University of Hong Kong in 1985, and the Ph.D. degree from the University of Strathclyde in 1993.

He is now an Associate Professor in the Department of Electrical Engineering, the Hong Kong Polytechnic University. His current research interests include power market reform and simulation, energy policy for sustainable and smart grids development.

Dr. Ngan is a Fellow IET, Fellow HKIE, and SMIEEE. He served as Chairman of the IEEE Hong Kong Section in 2003 and 2004.