Probabilistic Day-Ahead Load Forecast Using Quantile Regression Forests

Ali Lahouar, Amal Mejri and Jaleleddine Ben Hadj Slama
Laboratory of Advanced Technology and Intelligent Systems
National Engineering School of Sousse, BP 264, Sousse Erriadh 4023, University of Sousse, Tunisia
Email: ali.lahouar@gmail.com, elmejri.amal@gmail.com, bhslama@yahoo.fr

***Abstract*—Load forecast is one of the most important tasks in modern and smart grids. With the integration of renewable intermittent sources and the adoption of demand response strategies, an accurate short-term prediction becomes mandatory. Modern forecast approaches do not merely estimate future values, but provide also confidence intervals with different widths and probabilities. Therefore, this paper proposes a probabilistic day-ahead load forecast approach based on quantile regression forests. Quantile regression forests are extensions to random forests that provide confidence intervals instead of single points. The forecaster inputs are chosen according to measures of correlation and importance, profile analysis and wavelet decomposition of load curves. Several tests are performed using real data sets from the Ontario market. The results reflect the accuracy and the effectiveness of the proposed model under different circumstances.**

***Index Terms*—Load forecast, day-ahead, feature selection, quantile regression forest, probabilistic forecast, confidence intervals.**

I. Introduction

Due to the stochastic and uncertain nature of load, an accurate load forecasting is still a great challenging problem. Load prediction becomes a center of interest for several researchers, especially after the integration of renewable energy sources into grids. Indeed, intermittent energy sources and demand side management make the load forecast more challenging. However, most of the load forecast techniques suffer from lack of performance and accuracy. Research is still looking for the best and the most effective method to ensure balance between demand and supply in electricity markets, and to maintain the grid stability.

Based on time scale, load forecasting can be classified into three categories: short, medium, and long-term forecast. Short-term load forecast has a time horizon ranging from few minutes to few weeks ahead. Accurate short-term prediction is very useful for economic dispatch. The mid-term forecast horizon ranges from one month to two years, and it is mandatory for maintenance scheduling. Long-term forecast is sometimes extended to 20 years ahead or more. It is aimed at system expansion planning [1]. This paper will focus only on the short term, and specifically day-ahead forecast.

In the literature, a variety of methods have been proposed in order to address forecast accuracy issues. These methods are classified into three categories: statistical, artificial intelligence, and hybrid. Statistical methods use mathematical models that depend on historical load data and extra variables affecting the future load. They try to establish equations that match these quantities to future load values. This family of methods includes the well-known Box-Jenkins models; autoregressive integrated moving average (ARIMA) [2], autoregressive moving average (ARMA) [3] and autoregressive moving average with exogenous variables (ARMAX) [4]. Statistical methods include also exponential smoothing [5] and Kalman filter [6]. These methods are simple to implement, but they are unable to update their prediction strategies or to learn the new trends of load patterns. Because of these limitations, artificial intelligence methods are introduced. Intelligent methods are black-box models with unknown internal dynamic, which is established after a training process or according to a set of fuzzy rules. They can handle the problem of nonlinearity existing in the load series. This family of methods includes artificial neural networks (ANN) [7], fuzzy logic [8], random forests (RF) [9] and support vector machines (SVM) [10]. One of the major drawbacks of this type of methods is the need for optimization and parameter tuning to guarantee a good accuracy. Hybridization is one of the solutions to overcome these drawbacks. Hybrid methods are combinations of two or more of previous methods. For example, neural networks may be combined with fuzzy logic and wavelets transform [11]. It is possible also to use a Bayesian neural network with k-means clustering algorithm and time series analysis [12]. Hybridizing fuzzy support vector machine and linear extrapolation is proposed likewise [13]. Nevertheless, all mentioned methods focus on point forecast. Hence, they do not provide information about any possible uncertainty or error margins. Probabilistic forecast and confidence intervals are proposed in this context.

After the 2014 edition of the Global Energy Forecasting Competition (GEFCom), researchers are paying more attention to probabilistic forecast, which provides prediction intervals with different probabilities in addition to a median value. The aim of this type of forecast is to provide confidence intervals (CI) that can handle the uncertainty associated with prediction results. These intervals are used to assess the predictor accuracy and they are indispensable in many fields of application. A confidence interval defines a margin of error; it does not estimate directly the point, but the chance to contain this point. Each CI has upper and lower bounds that define its borders. A practical methodology to generate probabilistic load predictions is to perform quantile regression averaging on sister forecasts [14]. Likewise, it is possible to use fuzzy time series for predicting intervals [15]. These approaches may be compared to regular point forecasts. A recent review, including several techniques of probabilistic electrical load forecast, can be found in the literature [16].

This paper proposes a day-ahead load forecasting method. The prediction is probabilistic; nine imbricated confidence intervals are provided in addition to a median value for each hour of the day. The load data are first analyzed with the aim to select the most appropriate inputs for the forecaster. The load curves are decomposed using wavelets in order to extract the different levels of periodicity. The inputs are chosen according to these decompositions, in addition to correlation and importance measures. Then, dedicated forecasters are built for each hour of the day, after a large training period and via an online learning procedure. The quantile forest (QF) is chosen to construct the forecasters. The QF has all the advantages of random forests; it is not parametric, immune to irrelevant inputs, has a built-in cross validation process, and has a proper importance measure of its inputs. In addition, it provides the required intervals. The remainder of the paper is organized as follows. Section II analyzes the load data and extracts the periodicity levels. Section III depicts the forecast strategies; choosing inputs and building forecasters. Section IV shows and discusses qualitative and quantitative results. Section V concludes the paper.

II. Data analysis

A. Grid data and preprocessing

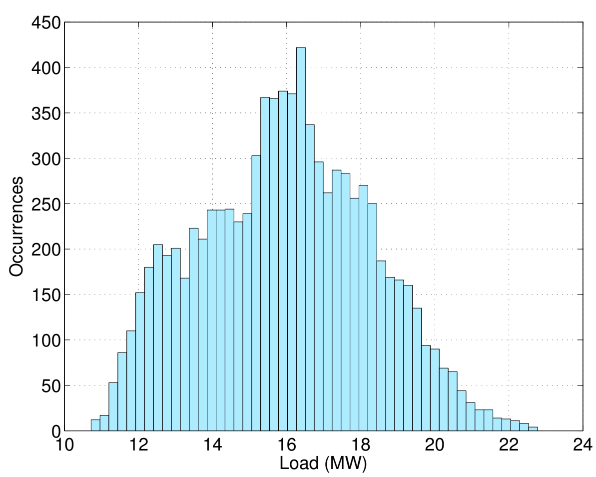

Fig. 1. Distribution of Ontario hourly load throughout 2014

The Ontario market is chosen to assess the performance of the proposed method, since it is one of the most studied cases in literature, along with Pennsylvania-New Jersey-Maryland Interconnection (PJM) and Australian market. Ontario is one of the most volatile electricity markets in the world. It is characterized by real time pricing, which may affect the load demand to some extent. Let $L(d,h)$ be the hourly load time series, where d stands for the day, and h for the discrete time in hours. The $L(d,h)$ distribution at the Ontario market over the year 2014 is given in Fig. 1. Obviously, most values are between 11 and 22 MW. The shape of the smoothed histogram resembles to a Gaussian centered on 16 MW. There is no discontinuity or extreme values that may disturb the training process, and this is beneficial for the prediction machine. Generally, extreme values are considered outliers and they have nasty impact on the forecast quality. In order to conserve the dynamics of the time series, no preprocessing is applied to $L(d,h)$. In addition, day type and season indexes are appended to $L(d,h)$ as extra information, and they will be detailed later. Temperature was ignored since it has been already included in the season index. This paper uses the data of 2014 and 2015 for training and testing purposes.

B. Profile analysis

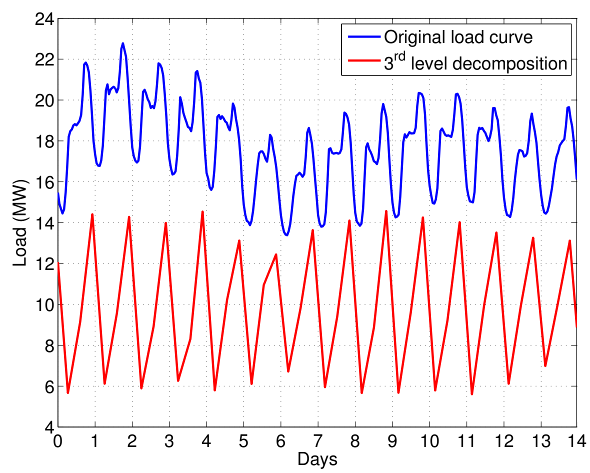

Fig. 2. Original and decomposed (3rd level) hourly load curves, from 6 to 19 January 2014

The profile analysis is the description of the load curve evolution through time for different horizon scales. Since load is biased by calendar effects, four levels of periodicity are supposed to appear; daily, weekly, seasonal and yearly. The yearly effect will be disgraced since it is only useful for long-term predictions. The daily periodicity may be extracted by wavelet packet decomposition [17]. The original load signal is then decomposed to the level 3, and the second signal (starting from the lowest frequency) is grabbed among the 8 generated signals. This signal appears in Fig. 2 just below the load curve, where it has been shifted to the top for clarity purpose (originally it was oscillating around zero). There is an obvious daily periodicity that should be considered while building the forecaster. Weekly periodicity needs further decomposition to be viewed, since it is located at lower frequencies. Therefore, the decomposition is performed to the sixth level, and the fourteenth signal is selected among the 64 generated signals. This signal is given in Fig. 3, where it is shifted to the top. Weekly periodicity appears in the form of difference between load demand on working days and weekends. This type of periodicity justifies the use of day type index.

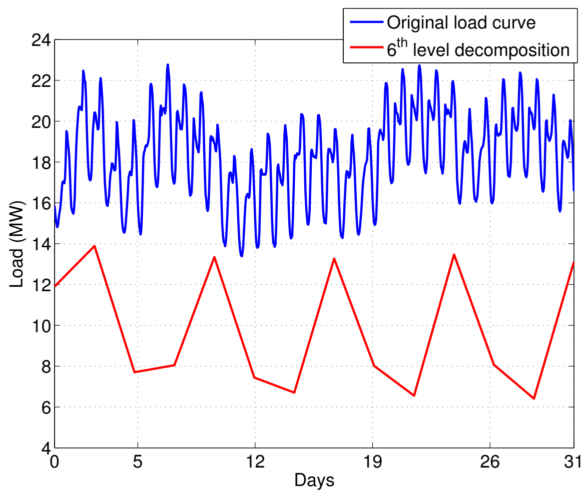

Fig. 3. Original and decomposed (6^{th} level) hourly load curves, from 1 to 31 January 2014

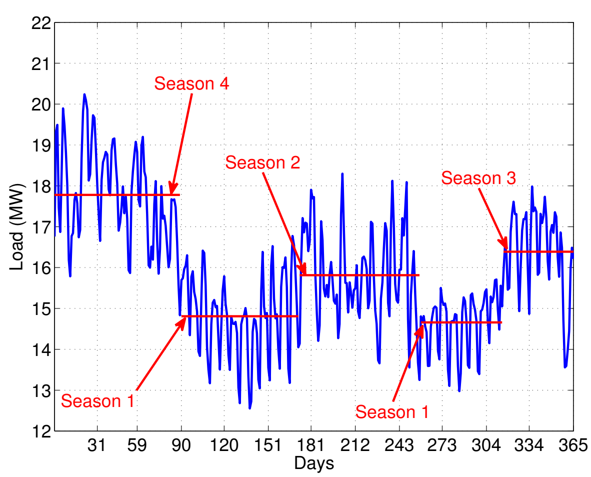

Fig. 4. Daily average load curve throughout 2014, with seasonal means

The seasonal effect cannot simply be extracted by wavelet decomposition. Indeed, the sub-sampling at each new level causes the signal points to be scarcer until they vanish completely. However, there are other methods to handle the seasonal periodicity. Fig. 4 shows the average daily load throughout the year 2014. The curve may be divided into 5 time periods with different average values (represented by horizontal lines). The second and fourth periods have almost the same average value, so they may be considered one single season. Based on this criterion, four seasons are defined, denoted season 1 to season 4, and sorted from lowest to highest average value. The seasonal effect is a direct consequence of temperature. Heating and cooling systems are the most consuming devices, so they have huge impact on the demand. This fact explains why the load increases in winter and summer and decreases in spring and autumn.

III. Forecast Strategy

A. Feature selection

The feature selection process is the art of choosing the most appropriate inputs for the forecaster. These inputs, called features or regressors, should be the most important factors driving the future load. The autocorrelation plot is one of the commonly used methods for selecting features. The autocorrelation function of load in January 2014 appears in Fig. 5. Maximum correlation measures are detected at lags of 24 and 168 hours, which asserts both daily and weekly periodicity. The similarity between the load curve and its 24-hours-lagged versions pushes towards building a dedicated forecaster for each hour. Each forecaster should provide $\hat{L}(d,h)$, where $\hat{L}(d,h)$ is the predicted load at hour h on day d, $h = 1, ..., 24$.

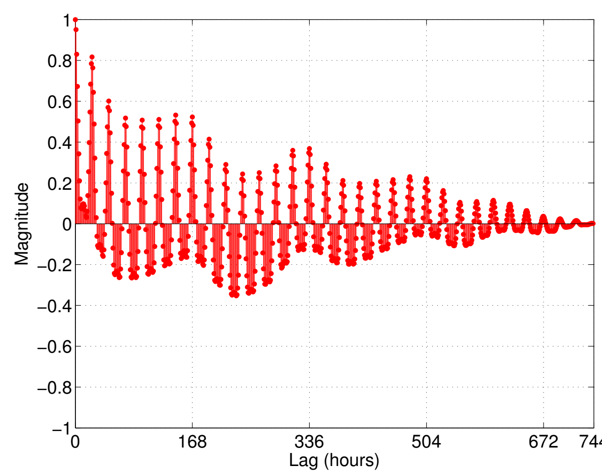

Fig. 5. Autocorrelation plot of the load time series in January 2014

In order to keep the daily periodicity, it is judicious to consider the load values of past days as inputs. Then, $L(d-1,h)$ and $L(d-2,h)$ are selected as regressors. According to Fig. 5, there is no need to get further samples at the same hour h, since similarity measures are almost equal beyond $d-2$. However, adding neighboring samples such as $L(d-1,h-1)$ and $L(d-1,h+1)$ may be beneficial. Obviously, $L(d,h-1)$ cannot be added, otherwise it will be hour-ahead forecast. The four features are then selected according to two directions; d direction and h direction. This particular selection encourages neighboring and tries to increase similarity. Until now, all chosen inputs are endogenous, which means extracted from the past output values. Exogenous inputs are also necessary, which are not historical measures of load. For example, a day index $Id(d)$ may be appended to the input vector. The index is a value assigned to the day according to its position in the week. For instance, 1000 is assigned for Mondays, 2000 for Tuesdays,... and 7000 for Sundays. This index helps to keep the weekly periodicity. Indeed, the forecaster can learn easily that load is slightly lower when the day index is 6000 or 7000, which means weekend. Finally, in order to reflect the seasonal periodicity, a season index $Is(d)$ is also added. This index will be the same for all the days d of the same season, according to Fig. 4. $Is(d)$ is equal to 0 in season 1, 10000 in season 2, 20000 in season 3, and 30000 in season 4. Huge values are intentionally chosen so as to give some importance to this effect.

TABLE I
IMPORTANCE AND CORRELATION WITH OUTPUT OF SIX INPUTS AT NOON, ON OCTOBER 1^{ST}, 2015

Input	Importance measure	Correlation with output
$Id(d)$	2.5417	-0.3350
$Is(d)$	1.1477	0.5318
$L(d-1,h)$	1.0369	0.7733
$L(d-2,h)$	0.4271	0.5747
$L(d-1,h-1)$	0.7932	0.7491
$L(d-1,h+1)$	0.9486	0.7810

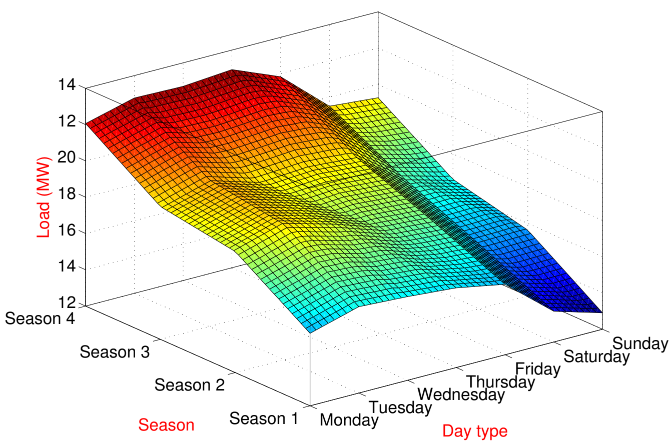

Fig. 6. Average daily load in 2014, in terms of day type and season

The six chosen inputs are given in Table I, along with measures of importance and correlation with the output, on 1^{st} October 2015 at noon. The importance measures are among intrinsic characteristics of the random forest [18]. These measures assess the importance of each input by computing a specific score. The more this score is high, the more the input is important. As it can be seen, the most important input is the day index $Id(d)$, followed by season index $Is(d)$ and past load $L(d-1,h)$, respectively. The maximum correlation is given by $L(d-1,h)$ and $L(d-1,h+1)$, while correlation with $Id(d)$ is negative. Negative correlation is expected, since load is supposed to decrease when the day index increases. Correlation with $Is(d)$ is positive, and this also expected. Indeed, the highest seasonal indexes are assigned to most consuming seasons. Fig. 6 shows a surface that fits the average daily load in 2014 to day type and season. The correlation with $Id(d)$ and $Is(d)$ are well justified by the surface shape. The load increases following the season direction and decreases following the day direction.

B. Building the forecasters

The forecasters are built using quantile regression forests [19]. The quantile forest is a generalization of the random forest (RF) used for regression. A random forest is an ensemble method that combines the prediction of several uncorrelated regression trees via an algorithm called bagging. It is then a learning machine that needs a dedicated training process. The RF has a number of advantages over other artificial intelligence machines. It is characterized by built-in cross validation and intrinsic measure of importance. In addition, it is immune to irrelevant inputs. However, it provides only one single output for each input vector. It is then used only for point forecast. In order to add confidence intervals for the prediction, the quantile forest (QF) is elaborated. The QF does not return the average response of all trees, but the full distribution of all responses. This distribution, which has the form of confidence intervals, is useful to handle the forecast uncertainties. In this paper, 9 confidence intervals are considered. The QF forecaster should then return 19 values for each input vector; one median value, and 2 bounds for each interval.

The followed learning process is online. It means that a new training is performed for each new day in the test period. Naturally, this training should be repeated 24 times to build the 24 forecasters of the day. For example, in order to predict the load on day d, all the period from $d-360$ until $d-1$ is chosen for building 24 forecasters. To forecast load on day $d+1$, the period from $d-359$ until d is chosen for building 24 new forecasters, and so on. The training period is sliding in order to learn the new load patterns and to limit transit problems between seasons. In this paper, four months are used for test. For each individual day within these months, a total training period of 360 days is needed.

IV. CASE STUDY

January, April, July and October 2015 are the chosen months for test. They are intentionally chosen across seasons in order to assess the forecast quality under different scenarios. Point forecast results are given by the median prediction value, whereas probabilistic forecast is performed through prediction intervals. Both median value and interval bounds are provided by the QF forecasters depicted in the previous section. Fig. 7 shows the day-ahead forecast results in four weeks, from the 9^{th} to the 15^{th} day of each month. In each case, a black curve represents the point forecast, while nine prediction intervals are given in the form of blue color gradient. The darkest blue zone represents the 10% prediction interval, which is supposed

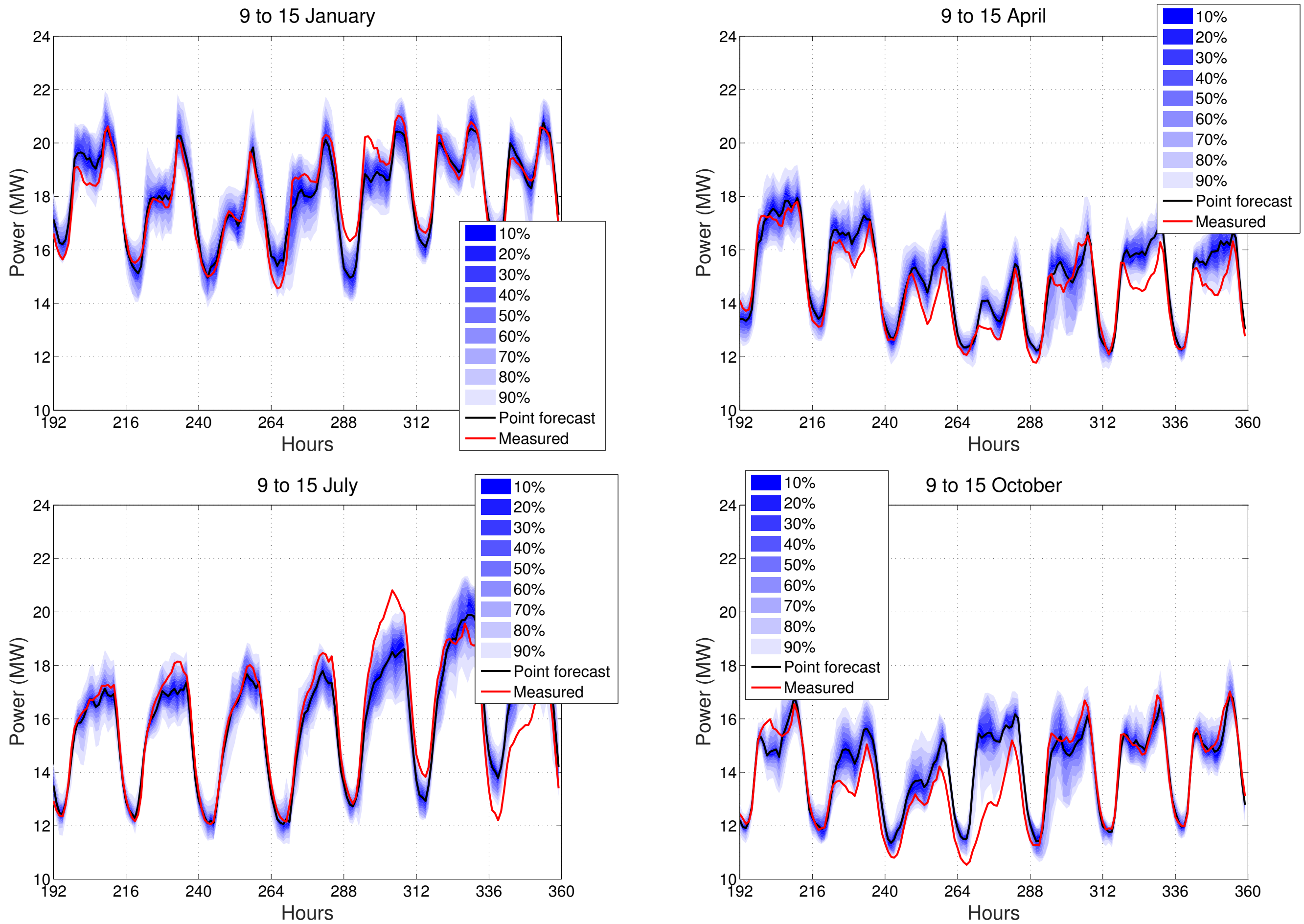

Fig. 7. Predicted confidence intervals and actual load curves throughout four test weeks

to contain 10% of measured points. It has the narrowest bounds. The subsequent intervals are ranging from 20% to 90%. Their bounds are getting larger and their colors are getting lighter as they are getting away from the median value.

Forecast results in January are pretty satisfying. Almost all measured points are inside bounds. Results in April are also within limits, although they are a bit overestimated. Generally, narrower intervals reflect better accuracy. However, prediction intervals on April 13$^{\text{th}}$ are somehow large. The same thing is observed on July 13$^{\text{th}}$ and 14$^{\text{th}}$, although the first four days were very accurate. Indeed, the forecaster predicted a huge variation on July 13$^{\text{th}}$, but it was not able to determine if it would be an increase or a decrease. Results in October are almost satisfying, despite some large intervals.

Qualitative assessment through curves is necessary, but it is not enough. Quantitative evaluation through error metrics is also required. For this purpose, three evaluation criteria are proposed: the mean absolute error (MAE), the root mean squared error (RMSE) and the mean absolute percentage error (MAPE). If the test period contains n days, these errors are defined by:

$$MAE = \frac{1}{24n}\sum_{d=1}^{n}\sum_{h=1}^{24}|\hat{L}(d,h) - L(d,h)| \tag{1}$$

$$RMSE = \sqrt{\frac{1}{24n}\sum_{d=1}^{n}\sum_{h=1}^{24}(\hat{L}(d,h) - L(d,h))^2} \tag{2}$$

$$MAPE = 100 \times \frac{1}{24n}\sum_{d=1}^{n}\sum_{h=1}^{24}\frac{|\hat{L}(d,h) - L(d,h)|}{L(d,h)} \tag{3}$$

The MAE is used to measure the average distance between predicted and actual quantities, through the absolute value to avoid any possible offset. The RMSE does the same thing but with a small difference; the error offset is avoided by squaring. The MAE and RMSE are expressed in terms of MW. The MAPE divides the absolute errors by actual load values in order to get a percentage. It provides then a normalized metric. All these criteria apply only to point forecast.

Table II shows the proposed metrics computed in four test months. Obviously, the RMSE is always a bit higher than MAE. The MAE does not exceed 1 MW, while the MAPE ranges from 3% to 5%. Nevertheless, these values are not significant if they are not compared with similar forecast methods. Therefore, the persistence (PER) and artificial neural networks are also proposed as forecasters. The persistence assumes simply that future load values are equal to actual values. The ANN is a feedforward network with one hidden layer of 10 neurons. According to Table II, the QF gives the best results in most cases. Although it is close to ANN in terms of accuracy, it provides lower errors especially in July and October. In addition, the QF has an advantage over ANN; it does not need to be tuned or optimized. However, the ANN has several parameters to adjust. Any bad choice of these parameters degrades significantly the forecast accuracy.

TABLE II
COMPARISON OF DIFFERENT FORECAST METHODS IN TERMS OF THREE PROPOSED CRITERIA

Criterion	Persistence	Quantile forest	Neural network
January			
MAE	0.8130	0.5953	0.5943
RMSE	1.0417	0.7653	0.7745
MAPE	4.6536	3.4320	3.4317
April			
MAE	0.7453	0.5406	0.5516
RMSE	1.0305	0.7039	0.7347
MAPE	5.1488	3.7543	3.8020
July			
MAE	0.8877	0.8744	0.9274
RMSE	1.1838	1.1292	1.2127
MAPE	5.3905	5.1923	5.5253
October			
MAE	0.5950	0.4351	0.4993
RMSE	0.8687	0.6445	0.6925
MAPE	4.1480	3.0999	3.5108

V. CONCLUSION

Electrical load forecast is nowadays a mandatory and challenging task for grid managers. Accurate short-term forecast in particular is extremely important for dispatch. In this study, a probabilistic load forecast method was proposed for this purpose. The quantile regression forests were chosen for building 24 dedicated forecasters for each hour of the day. The learning procedure was online with large training periods. The inputs were chosen according to correlation and importance measures, and after analyzing different levels of periodicity. Results were pretty satisfying whether under qualitative or quantitative assessment. The confidence intervals provided by quantile forests offered useful tolerance margins in case of errors. In addition, the proposed model showed its effectiveness and superiority once compared to similar traditional methods.

REFERENCES

[1] S. R. Khuntia, J. L. Rueda, and M. A. M. M. van der Meijden, "Forecasting the load of electrical power systems in mid- and long-term horizons: a review," *IET Generation, Transmission Distribution*, vol. 10, no. 16, pp. 3971–3977, 2016.

[2] C.-M. Lee and C.-N. Ko, "Short-term load forecasting using lifting scheme and {ARIMA} models," *Expert Systems with Applications*, vol. 38, no. 5, pp. 5902 – 5911, 2011.

[3] M. D. Simoes, M. C. Klotzle, A. C. F. Pinto, and L. L. Gomes, "Electricity distributor load forecasting with arma models," in *SSRN*, January 2013, pp. 1–14.

[4] C.-M. Huang, C.-J. Huang, and M.-L. Wang, "A particle swarm optimization to identifying the armax model for short-term load forecasting," *IEEE Transactions on Power Systems*, vol. 20, no. 2, pp. 1126–1133, May 2005.

[5] P. M. Macaira, R. C. Sousa, and F. L. C. Oliveira, "Forecasting brazil's electricity consumption with pegels exponential smoothing techniques," *IEEE Latin America Transactions*, vol. 14, no. 3, pp. 1252–1258, March 2016.

[6] H. Al-Hamadi and S. Soliman, "Short-term electric load forecasting based on kalman filtering algorithm with moving window weather and load model," *Electric Power Systems Research*, vol. 68, no. 1, pp. 47 – 59, 2004.

[7] P. Ray, D. P. Mishra, and R. K. Lenka, "Short term load forecasting by artificial neural network," in *2016 International Conference on Next Generation Intelligent Systems (ICNGIS)*, Sept 2016, pp. 1–6.

[8] M. F. I. Khamis, Z. Baharudin, N. H. Hamid, M. F. Abdullah, and F. T. Nordin, "Short term load forecasting for small scale power system using fuzzy logic," in *2011 Fourth International Conference on Modeling, Simulation and Applied Optimization*, April 2011.

[9] A. Lahouar and J. B. H. Slama, "Day-ahead load forecast using random forest and expert input selection," *Energy Conversion and Management*, vol. 103, pp. 1040 – 1051, 2015.

[10] M. S. Li, J. L. Wu, T. Y. Ji, Q. H. Wu, and L. Zhu, "Short-term load forecasting using support vector regression-based local predictor," in *2015 IEEE Power Energy Society General Meeting*, July 2015, pp. 1–5.

[11] A. S. Pandey, D. Singh, and S. K. Sinha, "Intelligent hybrid wavelet models for short-term load forecasting," *IEEE Transactions on Power Systems*, vol. 25, no. 3, pp. 1266–1273, Aug 2010.

[12] M. Ghofrani, D. Carson, and M. Ghayekhloo, "Hybrid clustering-time series-bayesian neural network short-term load forecasting method," in *2016 North American Power Symposium (NAPS)*, Sept 2016, pp. 1–5.

[13] X. Jiang, X. H. Liu, and R. Gao, "Hybrid load forecasting method based on fuzzy support vector machine and linear extrapolation," in *Proceedings of the 10th World Congress on Intelligent Control and Automation*, July 2012, pp. 2431–2435.

[14] B. Liu, J. Nowotarski, T. Hong, and R. Weron, "Probabilistic load forecasting via quantile regression averaging on sister forecasts," *IEEE Transactions on Smart Grid*, vol. 8, no. 2, pp. 730–737, March 2017.

[15] P. C. L. Silva, H. J. Sadaei, and F. G. Guimares, "Interval forecasting with fuzzy time series," in *2016 IEEE Symposium Series on Computational Intelligence (SSCI)*, Dec 2016, pp. 1–8.

[16] T. Hong and S. Fan, "Probabilistic electric load forecasting: A tutorial review," *International Journal of Forecasting*, vol. 32, no. 3, pp. 914 – 938, 2016.

[17] M. V. Wickerhauser, "Lectures on wavelet packet algorithms," in *Lecture notes, INRIA*, 1991, pp. 31–99.

[18] A. Liaw and M. Wiener, "Classification and regression by randomforest," *R News*, vol. 2, no. 3, pp. 18–22, 2002.

[19] N. Meinshausen, "Quantile regression forests," *J. Mach. Learn. Res.*, vol. 7, pp. 983–999, dec 2006.