2019 IEEE 3rd International Electrical and Energy Conference (CIEEC) 978-1-7281-1675-4/20/$31.00 ©2020 IEEE 10.1109/CIEEC47146.2019.CIEEC-2019284

Fill Missing Data for Wind Farms Using Long Short-Term Memory Based Recurrent Neural Network

Tie Li, Junci Tang, Feng Jiang, Xiaopeng Xu
State Grid Liaoning Electric Power Supply Company
Shenyang Liaoning, China

Chunzhu Li, Jiawen Bai, Tao Ding
Xi'an Jiaotong University
School of Electrical Engineering
Shaanxi Xi'an, China
tding15@mail.xjtu.edu.cn

Abstract—**Due to the uncertainty and volatility of wind energy resources, its large-scale consumption in power gird needs to be based on accurate prediction of output. This puts high demands on the integrity and accuracy of historical wind power data. However, in many wind farms, data loss due to equipment failure or human factors is common, which has a negative impact on wind power forecasting. In this paper, a Long Short-term Memory (LSTM) strategy is incorporated in the recurrent neural network (RNN) to set up a prediction model and fill the wind power missing data, which behaves better than the traditional RNN methods. The case of this paper uses the historical wind power data of Liaoning Province, which obtains the ideal results, proving the validity of the proposed model and method.**

Index Terms—**missing data, wind power, data filling, Long Short-term Memory (LSTM), Recurrent Neural Network (RNN)**

I. INTRODUCTION

Due to the shortage of fossil fuels, the rising of power demand, and severe environmental problems, wind power, as a clean and renewable energy source, has developed rapidly in recent decades and meantime its installed capacity has increased expeditiously. As a result of the strong randomness and intermittent of wind power, the output fluctuates drastically, which brings a challenge to the operation and scheduling of power systems [1-2]. In order to ensure the stability and safety of the power grid, it is essential to predict the wind power output by analyzing the historical data of wind farm.

Wind power output data is of great significance for the prediction and consumption of wind power. However, wind farms are often affected by some physical or none physical factors when recording data [3], resulting in missing data problem such as the failure of data observing device. Therefore, it is difficult to ensure the continuity of historical data, which causes a prominent negative impact on wind power forecasting. Therefore, the filling of missing data for wind farms has become one of the key issues in wind power forecasting and consumption at present.

Traditional missing data filling methods in power system mainly include average interpolation method, standard power curve method, regression, nearest neighbor method and many other simple methods [4-7]. For the specific situation of wind farm, in recent years, there has been a certain degree of research on how to deal with the missing data. Some of these studies focus on how to deal with missing data on wind speeds in wind farms. Reference [8] establishes a vertically correlated echelon model (CEM), which focus on comparing the level of wind speed. Reference [9] uses the least squares support vector machines (LSSVM) model to build wind speed data model, involving a CLPSO which has a strong global search capability. According to the analysis of the wind turbine model, reference [10] proposed traditional mathematical ways are hard to establish the wind model. While the model based on BP neural network performs better. Reference [11] compares the efficiency of two popular methods in missing data treatment. The others have studied how to deal with the lack of wind farm output power data. Reference [12] estimates the missing data by establishing data interpolation and regression models. The method is proved to be effective by real wind farms.

With the continuous expansion of data scale and the rapid improvement of data processing capabilities, many complex questions like [13] can be solved owing to this. Some deep learning methods can also be used to process and analyze data. As a special neural network structure with memory function, recurrent neural network (RNN) has been widely researched and applied in situations of continuous data, including machine translation, similarity calculation, speech recognition, natural language processing, commodity recommendation and other application fields. For power system, RNN also has a common usage in components, software and computing methods. For damping the oscillations of the multi-machines power system, [14] introduces a stabilizer for multi-machines using the tool of RNN. And the new-established stabilizer performs better. In [15], a neural networks library which is used for RSCAD is applied in power system. The results show that it can be an important tool in the future application. Reference [16] proposes a new method which consists of

This work is supposed by the Technology Project of SGCC-Research on "The aid decision making and visualization for real-time integration and dispatch for renewable energy based on the cloud platform" Led by State Grid Liaoning Electric Power Supply Company (SGLNDKOODWJS1900039).

RNN to model power system load. This method contributes to establishing accurate load model. Reference [17] proposes the use of RNN in power quality conditioners. Reference [18] introduces an impedance identification method using RNN. Compared with the traditional one, this new method shortens the test time greatly.

However, there are some problems with the RNN algorithm. For instance, the gradient disappearance or the gradient explosion. Therefore, a series of improved algorithms have emerged. Long Short-term Memory (LSTM) is an improved method of RNN. Based on the RNN, LSTM adds filtering to past states, so that you can choose the states which have more impact on the current, rather than simply selecting the most recent state. Recently, the LSTM method are basically used in prediction of renewable energy generation. In some study, researchers focus more on the forecast of solar power generation [19], and wind power generation [20]. Reference [21] introduces an assessment model of renewable energy accommodation capacity based on LSTM under the framework of TensorFlow.

In this study we will develop the Long Short-term Memory (LSTM) based recurrent neural network, which can construct a time series prediction model with long-term prediction ability for wind power data recovery.

The paper is organized as follows. Section II introduces the theory of RNN and the mathematical structure of LSTM. Section III shows the consequence and the analysis of the method using the practical wind power data from Liaoning province in China. Conclusions are drawn in Section IV.

II. Theory Of RNN And LSTM

Nowadays neural networks are widely used in the data processing. However, there are many kinds of neural networks and their structures determine that they have different usages. Convolutional neural network (CNN) uses the convolution core as an intermediary and shows a good performance in identification and simulation. However, CNN can't handle changes in time series which drives researchers to invent recurrent neural network (RNN), whose structure is listed as follows.

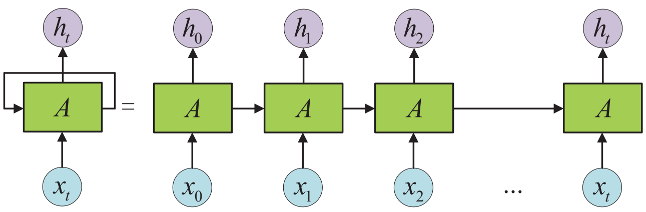

Fig.1. A simplified structure of RNN

In Fig.1, A is on the behalf of neural network. x_t is the input of while h_t represents the output at time t, respectively. What's more, both variables are vectors. The information in the current round will be passed to the next round in RNN through the connections between A. This structure endows RNN strong recognition ability. Therefore, RNN is also widely used to deal with time serialized applications such as natural language understanding (including speech recognition, translation, handwriting recognition), image and video recognition.

Although many of the advantages of RNN have been mentioned above, RNN still has a big flaw. As a memory neural network, RNN can't remember features very long time ago. In other words, RNN can't get precise prediction when this prediction is mainly based on very early input.

From Fig.1 we can inter a simplified equation, shown as

$$h_t = W \cdot h_{t-1} + \mathrm{U} \cdot x_t + b \tag{1}$$

In equation (1), if the absolute value of W is smaller than 1, the coefficient of the early h is infinitesimal, which is known as gradient diminish. When the absolute value of W is bigger than 1, the coefficient of the early h is close to infinity, which is known as gradient explosion. This is the reason why RNN can't deal with long-term prediction [22].

However, such shortcomings are intolerable in the missing data filling of wind farms' power output. Wind farm power generation data has a good continuity, which means that it usually does not change too much in a short time. This feature determines that timing prediction models such as RNN are well suited for use in the missing wind power data recovery. Meanwhile, the regularity cycle of wind power generation data tends to be relatively long due to the uncertainty of climate change in the short term. Therefore, we need to use a time series prediction model with long-term prediction ability for wind power data recovery. Finally, we use another variant of RNN which is known as LSTM to deal with wind farm power data prediction problem.

LSTM is a special variant of RNN. It is first introduced in [23]. The researchers design LSTM to solve the problem of gradient explosion and gradient diminish. As we have demonstrated in (1), when RNN is making long-term predictions the coefficient of early parameters is likely to become infinite or infinitesimal. But LSTM can solve this problem commendably. The following part will discuss the structure and design of LSTM and the reason why LSTM has a good performance in long-term learning.

LSTM also has repeating neural network modules like other RNN, but it's much more complicate than traditional RNNs. However, traditional RNNs usually have only one layer in one neural part while LSTM usually have more than four layers, whose detail is shown in Fig.2.

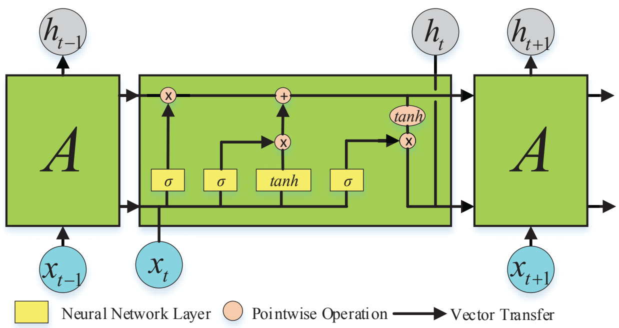

Fig.2. A simplified structure of LSTM

The prediction of LSTM is based on the past information, and LSTM should pick up the important information that can strongly affect the results. The components that play this role in LSTM are called layers. The coefficient of the layer determines how much this information can affect the further results. The out value of layer is always strictly between 0 and 1, the bigger output means this input has greater influence in the prediction result. The gate in the LSTM will combine the results of layers through some mathematical computations like convolution, product or some nonlinear functions. LSTM will continuously adjust the coefficients of layers and gates by using the known information and predictions the train the model. The model prediction will be more accurate with the process of training. The special repeating part of LSTM shown in Fig.3 assures that the old aged information can still influence the future data.

Firstly, LSTM will filter the past information and the information in this period and leave the useful information for the prediction. LSTM will check the input information h_{t-1} and x_t, then we can get a result which will be between 0 and 1 and we use it to update state C_{t-1}. The mathematical expression of upper parts is shown as

$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f) \tag{2}$$

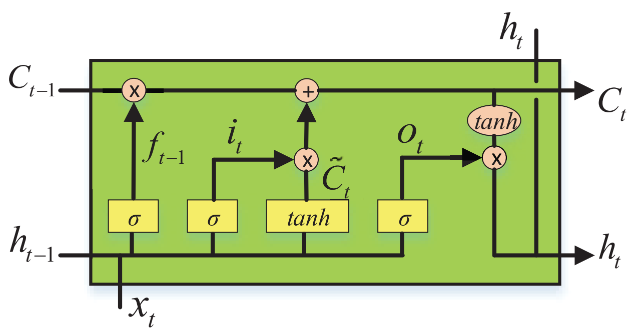

Fig.3. Details of LSTM's repeating module

Then we should choose some useful information and add them the gate for memorization. The information usually contains linear and nonlinear parts, represented respectively by equations (3) and (4)

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i) \tag{3}$$

$$\tilde{C}_t = \tanh(W_C \cdot [h_{t-1}, x_t] + b_C) \tag{4}$$

We have got enough old information so far so that in the next step, we will update the status based on the learning coefficient of information, which is depicted as

$$C_t = f_t \cdot C_{t-1} + i_t \cdot \tilde{C}_t \tag{5}$$

Finally, we get an output prediction based on the status C_t and the information we have at this time. What's more, we should pass this output to the next round for the next prediction. So we have

$$o_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o) \tag{6}$$

$$h_t = o_t \cdot \tanh(C_t) \tag{7}$$

The overview of the missing data recovery framework is shown in Fig.4. The program first receives the candidate data set and then cleans and processes the wind power data. Furthermore, the data is used to train the LSTM model, and finally the trained model is employed for missing data recovery. With the help of LSTM and the large number of recorded wind farm power generation data, we can recover the missing data with relative accuracy. In the following part we will use LSTM to recover missing wind power data and analyze the effectiveness of the model.

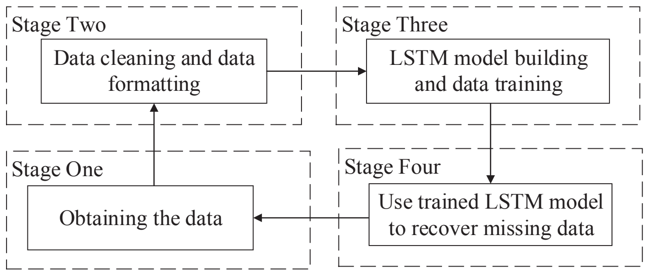

Fig.4. Overview of missing data recovery model

III. Case Study

The wind power data of Liaoning province is used for prediction in this part. The model is carried out using Python 3 on a work station with a two-core 3.20GHz CPU and 16.0 GB RAM memory. The programming was built and solved with the help of keras which is a python toolbox. The wind power data we used in this case is recorded per 15 minutes.

To analyze the effectiveness of the LSTM model, we use historical wind active power data to train the model [24-28]. The aim of our model is to minimize the mean squared error. For each case in our model we only lookback on the last one data point. We simulated the results using two different training set sizes. In these two simulations, we use 200 or 16338 points to train the LSTM model, respectively. After that, we test the effectiveness of the model by giving previous data and fill the output through prediction at this time. Then, compare the prediction and original data. For each case, we test 96 time checkpoints (exactly one day).

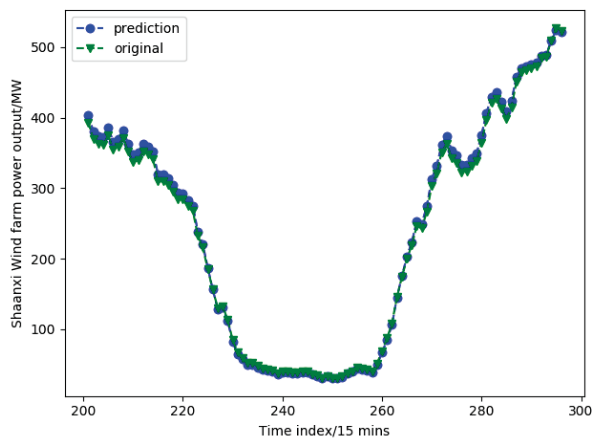

Fig.5. Prediction when using 200 data points for training

TABLE I PART OF DATA IN FIG.5

Time	Prediction	Original	Time	Prediction	Original
221	283.22	274.91	231	64.69	66.94
222	275.44	267.56	232	57.70	59.76
223	238.46	232.93	233	50.06	51.82
224	220.16	215.88	234	50.19	51.96
225	187.25	185.23	235	46.36	47.93
226	156.37	156.30	236	43.29	44.69
227	129.08	130.41	237	41.37	42.66
228	131.61	132.83	238	40.15	41.37
229	111.47	113.44	239	36.27	37.23
230	82.62	85.07	240	38.85	39.99

The prediction and original data are show in Fig.5 and to make it clear we show part of the data in

TABLE I. For the whole test dataset, the maximum of error is 10.86 and the mean squared error is 47.18. The prediction shows better accuracy when the data we need to predict is relatively small. The maximum error rate shows that the model is very effective.

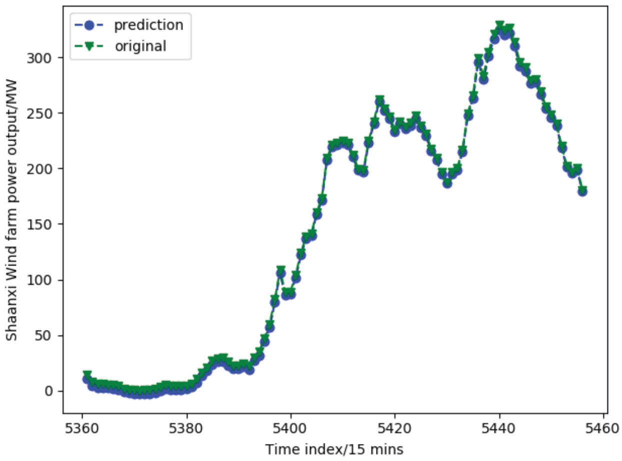

Fig.6. Prediction when using 5360 data points for training

Furthermore, we use 5360 data points for training. The prediction and original data is show in Fig.7 and to make it clear we show part of the data in TABLE III. For the whole test dataset, the maximum of error is 4.47 and the mean squared error is 7.80. The prediction result shows that using more data for training can significantly improve the model's accuracy.

TABLE II PART OF DATA IN FIG.7

Time	Prediction	Original	Time	Prediction	Original
5361	1.14	4.56	5371	19.40	22.74
5362	3.11	6.53	5372	21.35	24.68
5363	7.27	10.67	5373	19.01	22.36
5364	13.13	16.50	5374	26.52	29.80
5365	17.60	20.95	5375	31.81	35.04
5366	23.68	26.99	5376	43.88	46.99
5367	25.87	29.16	5377	56.88	59.82
5368	26.03	29.32	5378	79.47	82.08
5369	22.87	26.18	5379	105.99	108.21
5370	19.54	22.88	5380	85.83	88.35

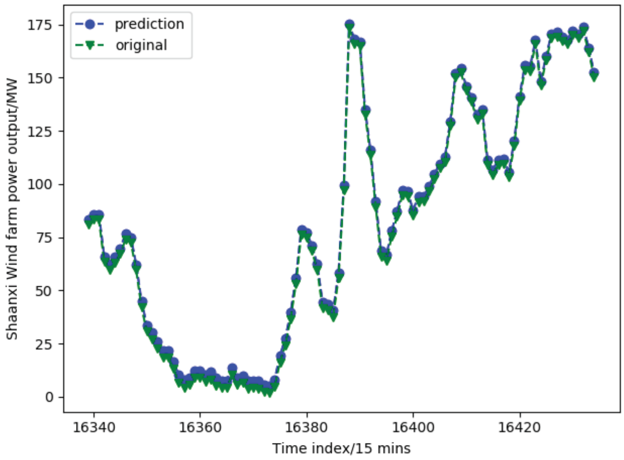

Fig.7. Prediction when using 16338 data points for training

Then, we use 16338 data points for training. The prediction and original data is show in Fig.7 and to make it clear we show part of the data in TABLE III. For the whole test dataset, the maximum of error is 3.48 and the mean squared error is 7.20. The prediction result shows that using more data for training can improve the model's accuracy. But when the amount of training data reaches a certain level, the improvement effect is not so obvious.

TABLE III PART OF DATA IN FIG.7

Time	Prediction	Original	Time	Prediction	Original
16349	62.17	59.45	16359	8.85	5.43
16350	44.87	41.95	16360	12.41	9.04
16351	33.70	30.63	16361	12.41	9.04
16352	30.24	27.12	16362	10.53	7.14
16353	26.00	22.83	16363	11.55	8.17
16354	21.65	18.41	16364	8.74	5.32
16355	21.52	18.29	16365	7.70	4.26
16356	16.57	13.27	16366	7.61	4.17
16357	10.10	6.70	16367	13.52	10.17
16358	7.77	4.34	16368	8.84	5.42

IV. CONCLUSION

This paper proposed a LSTM method for the missing data recovery of wind power generation. In this paper, we discuss the difference between RNN and LSTM. Then, the theory of LSTM is introduced to explain why LSTM has a good performance in long-term learning.

Finally, three cases using different size of training data are analyzed in the case study part. The results of these cases demonstrate that the LSTM model can give a prediction with excellent accuracy. When the data we need to predict is relatively small, the absolute error is also smaller but the relative error might be greater than other data. What's more, using a larger training set can significantly increase the accuracy of the model. Meanwhile, when the size of the training set reaches a certain level, increasing the size of the set has no great effect on the accuracy of prediction. The results confirm that using an appropriately sized training set can reduce training time while achieving not bad training accuracy.

REFERENCES

[1] T. Ding, Z. Wu, J. Lv and et al, “Robust Co-Optimization to Energy and Ancillary Service Joint Dispatch Considering Wind Power Uncertainties in Real-Time Electricity Markets, IEEE Transactions on Sustainable Energy,” vol. 7, no. 4, pp. 1547-1557, 2016.

[2] T. Ding, R. Bo, H. Sun and et al, “A Robust Two-Level Coordinated Static Voltage Security Region for Centrally Integrated Wind Farms,” IEEE Transactions on Smart Grid, vol. 7, no. 1, pp. 460-470, 2016.

[3] Hocaoglu, F. O. and Kurban, M. "The Effect of Missing Wind Speed Data on Wind Power Estimation. " International Conference on Intelligent Data Engineering & Automated Learning. Springer-Verlag, 2007.

[4] Z. G. Liu, Q. Pan, J. Dezert, A. Martin, “Adaptive imputation of missing values for incomplete pattern classification,” Pattern Recognition, vol. 52, pp. 85-95, 2016.

[5] L. Li, Y. Lin. “Short-term wind power forecasting based on an improved persistence approach”, Transactions of the CASE, 2010, 26(12):182-187.

[6] T. Ding, C. Li, C. Chen and et al, “Risk Assessment Based Long-Term Transmission System Hardening under Prior Probabilistic Information,” IET Generation, Transmission & Distribution, 2019, In-press.

[7] Z. Y. Lin. “Data collating skill for wind energy resource and wind measurement”. Guangdong Electric Power, 2003, 16 (5): 69-72.

[8] Q. Lin and J. Wang, "Vertically Correlated Echelon Model for the Interpolation of Missing Wind Speed Data," in IEEE Transactions on Sustainable Energy, vol. 5, no. 3, pp. 804-812, July 2014.

[9] X. Chen, B. Wang, M. Yu, J. Jin and W. Xu, "The interpolation of missing wind speed data based on optimized LSSVM model," 2016 IEEE 8th International Power Electronics and Motion Control Conference (IPEMC-ECCE Asia), Hefei, 2016, pp. 1448-1451.

[10] Y. Mao and M. Jian, "Data completing of missing wind power data based on adaptive BP neural network," 2016 International Conference on Probabilistic Methods Applied to Power Systems (PMAPS), Beijing, 2016, pp. 1-6.

[11] B. Lotfi, M. Mourad, M. B. Najiba and E. Mohamed, "Treatment methodology of erroneous and missing data in wind farm dataset," Eighth International Multi-Conference on Systems, Signals & Devices, Sousse, 2011, pp. 1-6.

[12] C. N. Sánchez, J. Enríquez-Zárate, R. Velázquez, M. Graff and S. Sassi, "Analysis of wind missing data for wind farms in Isthmus of Tehuantepec," 2018 IEEE International Autumn Meeting on Power, Electronics and Computing (ROPEC), Ixtapa, Mexico, 2018, pp. 1-6.

[13] T. Ding, J. Bai, P. Du and et al, “Rectangle Packing Problem for Battery Charging Dis-patch Considering Uninterrupted Discrete Charging Rate,” IEEE Transactions on Power Systems, 2019, In-press.

[14] C. Chen, T. Chen and J. Ou, "Power System Stabilizer Using a New Recurrent Neural Network for Multi-Machine," 2006 IEEE International Power and Energy Conference, Putra Jaya, 2006, pp. 68-72.

[15] B. Luitel and G. K. Venayagamoorthy, "Neural networks in RSCAD for intelligent real-time power system applications," 2013 IEEE Power & Energy Society General Meeting, Vancouver, BC, 2013, pp. 1-5.

[16] Wang Zhi-Qiang, Chen Xing-Qiong, Deng Chang-hong, Pan Zhang-da and Dong Chao, "The power system load modeling based on recurrent RBF neural network," 2007 International Power Engineering Conference (IPEC 2007), Singapore, 2007, pp. 910-915.

[17] U. S. Vadivu and B. K. Keshavan, "Power quality enhancement of UPQC connected WECS using FFA with RNN," 2017 IEEE International Conference on Environment and Electrical Engineering and 2017 IEEE Industrial and Commercial Power Systems Europe (EEEIC / I&CPS Europe), Milan, 2017, pp. 1-6.

[18] P. Xiao, G. K. Venayagamoorthy, K. A. Corzine and J. Huang, "Recurrent Neural Networks Based Impedance Measurement Technique for Power Electronic Systems," in IEEE Transactions on Power Electronics, vol. 25, no. 2, pp. 382-390, Feb. 2010.

[19] J. Zhang, Y. Chi and L. Xiao, "Solar Power Generation Forecast Based on LSTM," 2018 IEEE 9th International Conference on Software Engineering and Service Science (ICSESS), Beijing, China, 2018, pp. 869-872.

[20] G. Xu and L. Xia, "Short- Term Prediction of Wind Power Based on Adaptive LSTM," 2018 2nd IEEE Conference on Energy Internet and Energy System Integration (EI2), Beijing, 2018, pp. 1-5.

[21] C. Li, Y. Huang, C. Liu, Y. Wang, C. Dong and Z. Liang, "Research on Evaluation Method of Renewable Energy Accommodation Capability Based on LSTM," 2018 2nd IEEE Conference on Energy Internet and Energy System Integration (EI2), Beijing, 2018, pp. 1-5.

[22] Bengio Y, Simard P, Frasconi P, "Learning long-term dependencies with gradient descent is difficult, " IEEE Trans. Neural Networks, vol.5, pp. 157-166, Apr. 1994.

[23] S. Hochreiter and J. J. N. c. Schmidhuber, "Long short-term memory," Neural Computation, vol. 9, no. 8, pp. 1735-1780, March, 1997.

[24] T. Ding, R. Bo, F. Li and et al, “Interval Power Flow Analysis Using Linear Relaxation and Optimality-Based Bounds Tightening (OBBT) Methods,” IEEE Transactions on Power Systems, vol. 30, no. 1, pp. 177-188, 2015.

[25] T. Ding, Z. Wu, J. Lv and et al, “Robust Co-Optimization to Energy and Ancillary Service Joint Dispatch Considering Wind Power Uncertainties in Real-Time Electricity Markets, IEEE Transactions on Sustainable Energy,” vol. 7, no. 4, pp. 1547-1557, 2016.

[26] T. Ding, Q. Yang, X. Liu and et al, “Duality-Free Decomposition Based Data-Driven Stochastic Security-Constrained Unit Commitment, IEEE Transactions on Sustainable Energy,” 2019, in-press.

[27] T. Ding, S. Liu, W. Yuan and et al, “A Two-Stage Robust Reactive Power Optimization Considering Uncertain Wind Power Integration in Active Distribution Networks,” IEEE Transactions on Sustainable Energy, vol. 7, no. 1, pp. 301-311, 2016.

[28] T. Ding, Y. Hu and Z. Bie, “Multi-Stage Stochastic Programming With Nonanticipativity Constraints for Expansion of Combined Power and Natural Gas Systems,” IEEE Transactions on Power Systems, vol. 33, no. 1, pp. 317-328, 2018.