The Simulation of Noise Level Estimation for Energy Internet Load Data Prediction by Artificial Neural Network

Yangyang Ming
Department of Automation
Tsinghua University
Beijing, P.R.China
yitianxk@126.com

Junwei Cao
Beijing National Research Center for Infromation Science and Technology
Tsinghua University
Beijing,P.R.China
jcao@tsinghua.edu.cn

Qing Wang
New Smart City High-Quality Power Supply Joint Laboratory of China Southern Power Grid
Shenzhen Power Supply Co., Ltd.
Guangdong Shenzhen, P. R. China
csgwang@126.com

Huaying Zhang
New Smart City High-Quality Power Supply Joint Laboratory of China Southern Power Grid
Shenzhen Power Supply Co., Ltd.
Guangdong Shenzhen, P. R. China
zhyszpower@163.com

Xian Wu
New Smart City High-Quality Power Supply Joint Laboratory of China Southern Power Grid
Shenzhen Power Supply Co., Ltd.
Guangdong Shenzhen, P. R. China
wux19525@163.com

***Abstract*—In order to estimate the raw data (without noise) from the noisy data, we had proposed an ANN algorithm for simulating the relationships between history noisy data and corresponding raw data, which further made the noise level estimation. As the previous study was not detailed simulated, here we intensively re-estimate the algorithm's performance with different parameter set combinations, and get useful conclusions, which can help the application of the ANN algorithm in different situations. As preliminary research, we also proposed the future research directions, which may promote the performance of the algorithm to a further higher level.**

Keywords—Energy Internet; noisy data; noise level estimation; ANN

I. Introduction

As the step by step deeply advances of "Internet+" smart energy, and the general developing of new renewable energy, and the national proposed "double carbon plan", Energy Internet becomes the research hotspot in the electrical region in recent years [1-3]. Based on the ideas of "cyber physical system", and through source-network-load-storage distributed synergy, the energy running efficiency can be largely promoted, and realize the target of low carbon and environmental protection, which brings to the operator more sustainable revenues consisting of lowering running cost. So, Energy Internet is cited as one of the most potential applications in the future by scientists.

In order to reach high performance of power system like Energy Internet, precise load estimation is not unnecessary [4-6], and for further developing trend, noise level estimation can be a useful and potential service. Through this technique, we can detect the noise attack and/or equipment malfunction in real time, and work out useful polices to solve this problem as soon as possible.

Noise estimation is not a new topic, here only citing some typical application scenes (not time ordered). For example, D. D. Lee et.al use maximum likelihood estimation to detect noise in direction-of-arrival (DOA) estimation [7]. T. Yucek et.al use minimum mean square error to handle noise in OFDM frequency estimation [8]. N. Fan et.al use enhanced MCRA algorithm to handle noise in speech enhancement [9]. H. Yue et.al use CBM3D denoising method to realize image noise estimation [10]. P. Y. Kam et.al use Kalman filter to study different types of noise in sinusoid signal estimation [11]. A. Swami realizes multiplicative noise estimation through high order moments [12]. D. Starer et.al realize exponential signals estimation in noise through Newton iterative algorithm [13].

Through orthogonal decomposition, A. Mertins et.al realize impulsive noise estimation [14]. Through high order moments, Z. Zhang et.al realize colored noise estimation [15]. From the perspective of frequency domain, F. J. Vaquero-Caballero et.al realize perturbation-based linear and non-linear noise estimation [16]. In order to achieve generalized frequency division multiplexing, A. Mohammadian et.al realize joint channel and phase noise estimation in [17].

All issues form themselves efficient handling algorithms and proceedings, but most solutions are complex and need tedious logical deducing, which makes the handling delay unneglected, and could be a hard work when data quantity increases. As the ANN model is developing fast recently, which has high estimation performance and proper design complex and low running delay even for large data quantity, so we consider using ANN model for noise level estimation here.

The structure of this paper is as follows: chapter 1 introduces the background, chapter 2 introduces the application scene and proposed algorithm, chapter 3 analyzes the simulation result, chapter 4 proposes further research directions, chapter 5 makes the conclusion.

II. Algorithm Design Details

A. Application Scene

If we only know the noisy data and corresponding raw data which has the same probability distribution with the test data, and the raw value of test data need to be estimated, we can use the ANN algorithm to do the estimation quickly and accurately.

B. Proposed Algorithm

When the Energy Internet is running, there will be some underlying characters in raw load data, such as time periodicity and existing certain state transfer probability in data sequence. If we know the past data's distribution characters (in sequence), and only have noisy data to predict the noise level, we can utilize these inner characters to fulfill this task. As ANN is a useful predict model, and can learn almost any complexing function relationships without any detailed deducing, it can be used to estimate the raw data and noise level. As single data can't represent the developing trend of the data, data sequence is used for the ANN training and testing.

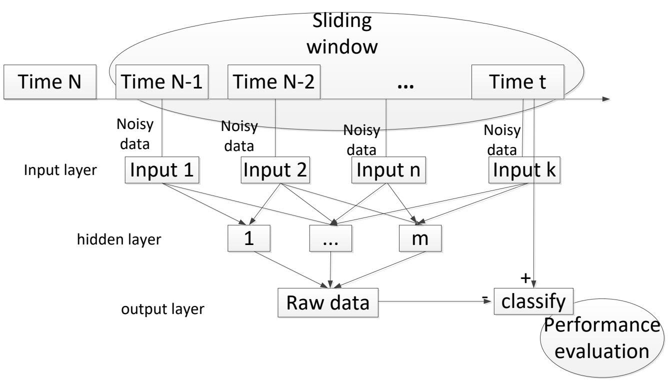

Fig. 1. Algorithm procedures

Follow [18], the algorithm step is as follows (shown in Fig. 1):

1. Using the load data and corresponding noisy data to statistically calculate the noise amplitude, we can classify the noise level, and obtain the noise range for every level.
2. We set the window length and move the window on the data sequence to fetch the data sequence along with time position, these data sequence will be used for model training.
3. When moving the time window, the nearest time position (the last element) of fetched data sequence is obtained, so we can also record the raw load data at that time position in simulation.
4. Using the fetched sequence as the input data, and the corresponding raw data as the output data, we train the ANN model using all data samples.
5. When new noisy data comes, we fetch the corresponding noisy data sequence belonging to the same time window of this new data, and use the ANN model to predict its corresponding raw value. We subtract the raw value with the noisy data, and classify the noise level using the range obtained in step1.
6. We calculate the average noise level index for some amount of new noisy data iterations.
7. We traverse the combination of different parameter set, and analyze the characters of these results, which can be used for further research and algorithm's performance prediction or promoting.

III. Simulation Result

We redo the simulation for [18], the executing steps are improved as follows.

1. As the load will be always more or equal than zero, we modify the noise value to be more or equal than zero, so no negative value is obtained, which can be expressed as:

$$noise\_amplitude = rand(1{,}100000) * index * max(max(data));$$

where the $noise\_amplitude$ is the amplitude of noise component, $rand$ is the uniform random distribution function for 100000 number of data, and $index$ represents the coefficient of noise amplitude.

2. Every result data is averaged for 10 times, increased from 4 times in previous simulation.
3. The simulation is executed for varying three parameters, such as hidden layer length for ANN, training data size and most importantly, the noise amplitude and so on. The size of simulation results is increased largely compared to initial simulation, which are more complete and more general, so more accurate results and clearer characters will be obtained, and useful policy can be deduced.
4. The algorithm results are compared with ANN with classify step algorithm, where it analyzes the reason why the performance of the classify algorithm is not as good as this one.
5. As another important factor, the time window length will be considered in further study.

```
net2=newff(minmax(P),[13,1], {'logsig','logsig' },'traingdx');
net2.trainParam.show=50;
net2.trainParam.lr=0.05;
net2.trainParam.mc=0.9;
net2.trainParam.goal=1e-3;
net2.trainparam.epochs = 500 ;
[net2,tr]=train(net2,P,T);
```

Fig. 2. ANN simulation

As mentioned before, we use the average noise level index to evaluate the performance of the related algorithms, which can be expressed as:

$$index = \frac{\sum abs(Lr - \overline{Lr})}{N * level} \quad (1)$$

where Lr represents real noise level, $\overline{Lr}$ represents estimated noise level, N presents data size, and $level$ represents classified total levels.

We program the procedure on the MATLAB, and use itself carried ANN tool box, the parameter setting is shown in Fig. 2.

The performance of the proposed algorithm can be statistically evaluated through main result analysis, mean value and variance value analysis, which is illustrated in part A, B and C as below.

A. Main Result Analysis

1) simulation 1

We fix the hidden layer length (13) at first, and vary the training data size and the noise amplitude, which include the parameter value vectors of [20000,30000,40000,50000] and [0.1,0.15,0.2,0.25,0.3,0.5] individually.

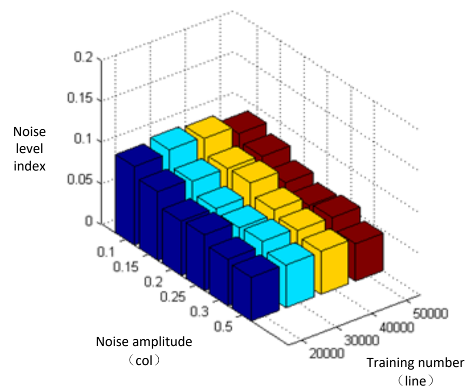

Fig. 3. Smulation result 1-1

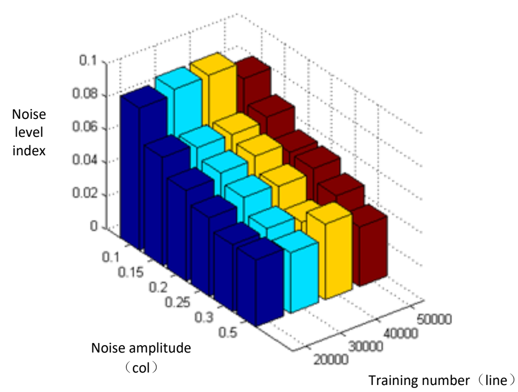

Fig. 4. Simulation result 1-2

The results are shown in Fig. 3 and Fig. 4, whose testing data samples will change from 50001 to 50100 in Fig. 3 and from 60001 to 60100 in Fig. 4. Other figures' results are choosing samples from 50001 to 50100.

From Fig. 3 we can see that, as training number increased and noise amplitude increased, the noise level index is decreased (performance increased). But as we can't guarantee to get the same results in other simulation scenes, we propose two hypothesis and test their validities.

Hypothesis 1: as training number increased, the classification performance will be improved (index value decreased).

Hypothesis 2: as noise amplitude increased, the classification performance will be improved (index value decreased).

As the related parameters of training number and noise amplitude are ordered from small to big value, so the test rule for the two hypotheses can be expressed as below, where N is the total data number, and we individually consider line and column index as below.

$$a_{ij} = \begin{cases} 1 & if\ V_{ij} < V_{i,j+1} \\ 0 & otherwise \end{cases}$$

$$line_e = \sum_i \sum_j a_{ij} / N \quad (2)$$

$$b_{ij} = \begin{cases} 1 & if\ V_{ij} < V_{i+1,j} \\ 0 & otherwise \end{cases}$$

$$col_e = \sum_i \sum_j b_{ij} / N \quad (3)$$

where V_{ij} means sample value at position i, j, $line_e$ represented the exception ratio whose data component didn't obey hypothesis 1 or hypothesis 2, in the line direction. And col_e represented the corresponding exception ratio in the column direction.

For Fig. 3, the count number of exception data in the line direction (training number) is 5, and the total data size is 24, so the line exception ratio is 5/24. And the column exception ratio (noise amplitude) is 4/24. Which verified both hypothesis 1 and hypothesis 2 in most conditions.

The line exception samples in Fig. 3 can be attributed to the random noise adding for noisy data and ANN model used, which means that the random distribution of the noise makes the mono decreasing laws can't be always obeyed and ANN is a statistical method inherently.

Also in Fig. 3, as noise amplitude increased, the data value becomes more randomly distributed, but at the same time, the domain range of every level increased, it can be rightly estimated more easily. So, with the interaction of the two factors, the hypothesis 2 is verified.

In Fig. 4, on one hand, we can see the exception ratio for line direction is 10/24, which is more than 40%, so the hypothesis 1 for the training number is not verified. But on the other hand, the exception ratio for the col direction is 1/24, which firmly verified the hypothesis 2 for noise amplitude.

In Fig. 4, as the training number increased, the reason why the corresponding index is not decreased may be explained by that the data sequence in the latter part of the data set is not very relevant with the test data sample as the previous part. So as the training number increased, the estimation performance is fluctuated even decreased for many results in that Figure, and many exception results emerged. Further promoting the training number to massive data may solve this problem in theory.

2) simulation 2

In simulation 2, we fixed the noise amplitude (0.3), and changed the training number and hidden layer length, whose vectors changed as [20000,30000,40000,50000] and [15,13,11,9,7] individually in Fig. 5.

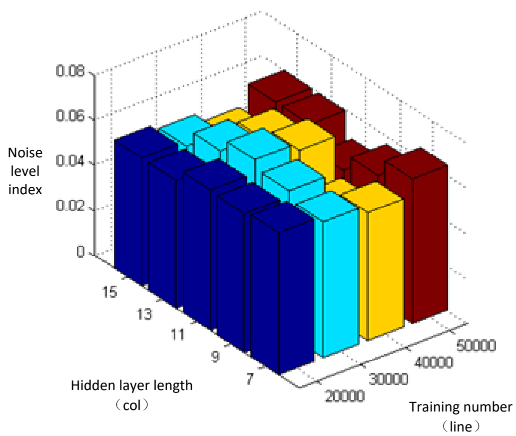

Fig. 5. Simulation result 2

From Fig. 5 we can see, in the line direction (training number from 20000 to 50000), the exception ratio is 6/20, so the hypothesis1 for the training number is not very positively verified.

3) simulation 3

In simulation 3, we fixed the training number (50000), and changed the noise amplitude and hidden layer length, with vectors [0.1, 0.15, 0.2, 0.25, 0.3, 0.5] and [15,13,11,9,7] individually. The result can be shown in Fig. 6.

From Fig. 6 we can see, in the line direction (noise amplitude from 0.1 to 0.5), the exception ratio can be 6/30, so the hypothesis 2 for noise amplitude can be barely verified.

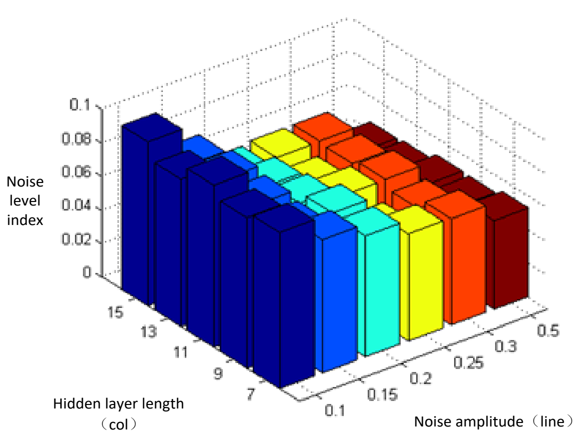

Fig. 6. Simulation result 3

B. *The Mean Value Result Analysis*

In some figures, we can see that the performance related to the hidden layer length is not mono increasing or mono decreasing, so it's not easy to find the optimal length, so we choose mean value and variance value sorting principles.

In Fig .4, the mean value in the training number dimension is: [0.0557,0.0566,0.0576,0.0554], which shows that training number 50000 has the least mean value.

In Fig .5, form the col direction, we can see that, if using first rank set and second rank set to choose the most proper hidden layer length, length 15 is the optimal one, followed by length 9. At the same time, if we use the mean value to choose the optimal hidden layer length. The result will be length 15 in the results array [0.053,0.0595,0.0611,0.0577,0.0612]. So, we can choose length 15 as the most proper system parameter.

In Fig. 6, if the mean value is used to choose the optimal hidden layer length, the most optimal one will be length 13 in the results [0.0659, 0.0651,0.0687,0.0657,0.0706], and the 2nd parameter will be length 9.

C. *The Variance Value Result Analysis*

The variance embodies the steady character of the algorithm, the smaller the variance, the steadier running of the algorithm. Here we calculate the variance of high layer data (already averaged).

The variance of the simulation results 1-1 (Fig .3) according to the training number is: [0.2672,0.3047,0.2711,0.2379], which shows that size 50000 has the least variance, at the same time, it has the least mean value. So size 50000 has the best performance in that scene.

The variance of the simulation results 1-2 (Fig .4) according to the training number is: [0.3100,0.3490,0.3284,0.2318], which shows that size 50000 has the least variance, which is agree with the mean value sort result.

The variance for the simulation results 2 (Fig .5) according to hidden layer length is: [0.1486,0.0532,0.7716,0.4913,0.1084], which shows that length 7 has the best performance in variance, and length 15 has the second least variance data, so if following the mean and variance principle at the same time, length 15 is a proper selection.

The variance for simulation results 3 (Fig .6) according to the hidden length layer is: [0.3301,0.2379,0.2531,0.2352, 0.2005], which shows that length 7 followed by length 9 are proper choices. Combined with above result, if using minimum variance as the choosing principle, length 7 is a proper choice.

D. *The Comparison Result Analysis*

At the same time of simulating this algorithm, we have designed a more complex algorithm (classify algorithm). In this algorithm, we first class the data sample sequences using modified single pass algorithm followed by k-means method, then we train one ANN model for every class using the data sequence belong to corresponding class. When predicting the noise level for new data, we classify the related data sequence to corresponding class by minimum distance principle, and use related ANN model to test the raw data, and predict its noise level.

The result is shown in Fig. 7, where ANN algorithm is the proposed algorithm in this simulation, and classify algorithm is the reference algorithm. From the Figure we can see, that although the algorithm with classifying method is more subtle and complexing, the result is not improved as expected. It can be deduced that, as we classify the data sequence, the sample number for every class is largely reduced, which may lead to overfit or not well trained for some ANN model with less data sample numbers, so the result may be suboptimal. Combined with complexity and performance evaluation, ANN algorithm is a proper selection for the scene of low or medium data size.

And at the same time, the result in Fig. 7 also verified the hypothesis 2 with noise amplitude decreasing law.

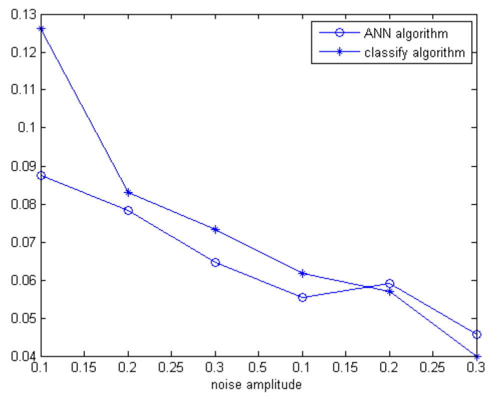

Fig. 7. Performance comparison with classify algorithm

E. Whole Performance Analysis

The highest performance result is 0.014 for single noise level index, and the lowest performance result is 0.116, which means that the parameter factors can have a notable effect in algorithm running, and the results are encouraging.

Intuitively, from the statistical view, if the length of time window coincides with the data characters, the prediction noise level will be more accurate, so the algorithm's performance can be improved, which will be detailed discussed in the following paper.

IV. FUTURE RESEARCH DIRECTIONS

1. We will test the algorithm on more complex probability distribution for noise component, such as Gaussian distribution and so on.
2. Massive training data size will be used for more subtle algorithm test, which can be 10× or 100× increased in data size.
3. Deep learning method may be used for ANN learning and testing.
4. With the increase of data size and programming complexity, cloud-edge synergy may be researched and deployed in Energy Internet.

V. CONCLUSION

In this paper, we intensively re-simulated the ANN algorithm for noise level estimation in the load data of Energy Internet. In order to research the characters of this algorithm in different parameter sets, we increase the simulation turns for every averaged result, and consider the combination of noise amplitude, training number and hidden layer length. From the simulation we can see, as the noise amplitude increases, the estimation error will decrease, and the training data size increase will not always prompt the estimation accuracy. Though the simulation result with corresponding mean value and variance value sorting, we can deeply learn the running characters of the ANN algorithm, and choose proper parameters to satisfy the real scene running condition of noise level estimation. As ANN is a common tool, it can be used in similar application scenes.

ACKNOWLEDGMENT

This paper is sponsored by Science and Technology Project of China Southern Power Grid (090000KK52190169/ SZKJXM2019669)

REFERENCES

[1] J. Cao, M. Yang, D. Zhang, Y. Ming, K. Meng, Z. Chen and C. Lin, "Energy Internet: an infrastructure for cyber-energy integration," Southern Power System Technology, vol.8, no.4, pp. 1-10, 2014.

[2] J. Wang, K. Meng, J. Cao, Z. Chen, L. Gao and C. Lin, "Information technology for Energy Internet: a survey," Journal of Computer Research and Development, vol.52, no.5, pp. 1109-1126, 2015.

[3] J. Cao, "Information system architecture for smart grids," Chinese Journal of Computers, vol.36, no.1, pp.143－167, 2013.

[4] L. Yin, F. Mo, Q. Wu and S. Xiong, "Research on short-term power load prediction based on deep learning", Proceedings of the 11th International Conference on Computer Engineering and Networks (CENet2021) Part I.,2021:164-170.DOI:10.26914/c.cnkihy.2021.045013.

[5] Y. Ming and J. Cao, "Electrical load prediction in Energy Internet via linear correlation coefficient approach," 2018 IEEE International Conference on Energy Internet (ICEI), 2018, pp. 157-162, doi: 10.1109/ICEI.2018.00036.

[6] N. Saengngam and U. Thonggumnead, "Predicting the medium-term electricity load demand of Thailand using the generalized estimating equation and the linear mixed effect model," 2015 12th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology (ECTI-CON), 2015, pp. 1-5, doi: 10.1109/ECTICon.2015.7206994.

[7] D. D. Lee and R. L. Kashyap, "Robust maximum likelihood bearing estimation in contaminated Gaussian noise," Fifth ASSP Workshop on Spectrum Estimation and Modeling, Rochester, NY, USA, 1990, pp. 104-108.

[8] T. Yucek and H. Arslan, "MMSE noise plus interference power estimation in adaptive OFDM systems," in IEEE Transactions on Vehicular Technology, vol. 56, no. 6, pp. 3857-3863, Nov. 2007.

[9] N. Fan, J. Rosca and R. Balan, "Speech noise estimation using enhanced minima controlled recursive averaging," 2007 IEEE International Conference on Acoustics, Speech and Signal Processing - ICASSP '07, Honolulu, HI, 2007, pp. IV-581-IV-584.

[10] H. Yue, J. Liu, J. Yang, T. Nguyen and C. Hou, "Image noise estimation and removal considering the bayer pattern of noise variance," 2017 IEEE International Conference on Image Processing (ICIP), Beijing, 2017, pp. 2976-2980.

[11] P. Y. Kam, Q. Wu and W. B. Lee, "Improved estimation of the frequency of a single sinusoid in noise using a new measurement model," The 8th International Conference on Communication Systems, 2002. ICCS 2002., Singapore, 2002, pp. 779-783 vol.2.

[12] A. Swami, "Multiplicative noise models: parameter estimation using cumulants," Fifth ASSP Workshop on Spectrum Estimation and Modeling, Rochester, NY, USA, 1990, pp. 212-216.

[13] D. Starer and A. Nehorai, "Maximum likelihood estimation of exponential signals in noise using a Newton algorithm," Fourth Annual ASSP Workshop on Spectrum Estimation and Modeling, Minneapolis, MN, USA, 1988, pp. 240-245.

[14] A. Mertins and N. Fliege, "Multiple parameter estimation of composite signals in uncharacterized impulsive noise," ICASSP-92: 1992 IEEE International Conference on Acoustics, Speech, and Signal Processing, San Francisco, CA, USA, 1992, pp. 557-560 vol.2.

[15] Z. Zhang, Y. Shi and H. Zhang, "An algorithm for joint angle and frequency estimation in colored noise," 2011 International Conference on Uncertainty Reasoning and Knowledge Engineering, Bali, 2011, pp. 219-222.

[16] F. J. Vaquero-Caballero, D. J. Ives, and S. J. Savory, “Perturbation-based frequency domain linear and nonlinear noise estimation”. Journal of Lightwave Technology, doi: 10.1109/JLT.2021.3112000..

[17] A. Mohammadian and C. Tellambura, “Joint channel and phase noise estimation and data detection for GFDM”. IEEE Open Journal of the Communications Society, 2021, pp.99.

[18] Y. Ming, J. Cao and H. Hua, “Noise Level estimation in Energy Internet based on Artificial Neural Network”, 2020 IEEE International Conference on Energy Internet (ICEI). IEEE, 2020.