

Achieving 100x Acceleration for N-1 Contingency Screening with Uncertain Scenarios using Deep Convolutional Neural Network

Yan Du, *Student Member, IEEE*, Fangxing Li, *Fellow, IEEE*,
Jiang Li, *Senior Member, IEEE*, Tongxin Zheng, *Senior Member IEEE*

Abstract—The increasing penetration of renewable energy makes the traditional N-1 contingency screening highly challenging when a large number of uncertain scenarios need to be combined with contingency screening. In this letter, a novel data-driven method, similar to image-processing technique, is proposed for accelerating N-1 contingency screening of power systems based on the deep convolutional neural network (CNN) method for calculating AC power flows under N-1 contingency and uncertain scenarios. Once the deep CNN is well trained, it has high generalization and works in a nearly computation-free fashion for unseen instances such as topological changes in the N-1 cases and uncertain renewable scenarios. The proposed deep CNN is implemented on several standard IEEE test systems to verify its accuracy and computational efficiency. The proposed study constitutes a solid demonstration of the considerable potential of the data-driven deep CNN in future online applications.

Index Terms—AC power flow, deep convolutional neural network (deep CNN), data-driven, image processing, N-1 contingency screening.

I. INTRODUCTION

RECENTLY, the increasing penetration of renewable energy into the bulk power system has aggravated the concern of system operation security under N-1 contingency. The main challenge for N-1 contingency screening under uncertainty is the extreme model complexity in case of large-scale power systems, combined with many uncertain scenarios. For instance, a traditional contingency screening for an N-branch system requires N power flow runs; however, once this is combined with M independent wind plants with 10 uncertain scenarios in each plant, contingency screening needs to be performed for $N \times 10^M$ power flow runs [1]. Even though it is well known that a traditional full-fledged N-1 contingency screening in the actual ISO operation takes only tens of seconds to complete, it will be unmanageable if many wind scenarios must be combined. Tremendous research efforts have been dedicated to accelerating computation speed for N-1 contingency screening in the literature [1-3].

The main issue with the traditional model-based methods is the huge number of power flow runs, which poses great computational burden that prevents their online applications even with proper model simplification. To address this issue, in this letter we present a novel data-driven method, similar to image processing, for accelerating N-1 contingency screening under multiple uncertain scenarios. The deep convolutional

neural network (deep CNN) is utilized as a regression tool for AC power flow calculation. Once the deep CNN is well trained, it is expected to have high generalization for unseen power flow cases and can improve the computational efficiency by over 100 times as compared with the conventional model-based methods in N-1 contingency screening. It should be noted that the solution can deal with topological change under N-1 contingency without the need of additional trainings.

II. DEEP CNN-BASED N-1 CONTINGENCY SCREENING

A. A brief on deep CNN

Deep convolutional neural network (deep CNN) is known for its strong automatic feature learning ability in processing data with a grid-like topology, e.g., image data. The term “deep” indicates that the deep CNN contains many hidden layers. Each hidden layer consists of a group of convolution kernels that extract features from the input via convolution operation. As the neural network goes deep, more features will be captured and analyzed. This hierarchical structure of deep CNN mimics the visual cortex system of human beings, which is the main reason for its success in image classification.

Deep CNN has an important feature, which is called sparse connectivity [4]. In image data, one pixel is closely related to its neighboring pixels, but is less related to the pixels that are far away. Hence in deep CNN, each output unit from the hidden layer is only connected to a square patch from the input that is geographically close to its location, and it discards the farther connections. In this way, the number of parameters that need to be trained is greatly reduced, which contributes to higher computational efficiency.

B. Mapping power system raw data to deep CNN input data

In the N-1 contingency screening problem, the deep CNN is proposed as a regression tool to automatically generate AC power flow (ACPF) results based on the known system parameters. Deep CNN is applicable for ACPF calculation because power system state variables, i.e., the bus voltage magnitudes and bus voltage angles, are also sparsely connected like image pixels. The power flow equations are as follows:

$$P_{inj,i} = v_i \sum_{j=1}^n v_j (g_{ij} \cos \theta_{ij} + b_{ij} \sin \theta_{ij}) \quad (1)$$

$$Q_{inj,i} = v_i \sum_{j=1}^n v_j (g_{ij} \sin \theta_{ij} - b_{ij} \cos \theta_{ij}) \quad (2)$$

In (1)-(2), it can be discovered that the bus voltage magnitudes (v_i and v_j) and angles (θ_i and θ_j), are mainly related to four known parameters including bus active power injection $P_{inj,i}$, bus reactive power injection $Q_{inj,i}$, and bus admittance g_{ij} and b_{ij} . As a result, for the ACPF calculation of a n -bus power

This work was financially supported in part by the CURENT research center, ISO New England, and NSF grant ECCS-1809458. Corresponding author: F. Li, e-mail: fli6@utk.edu.

Y. Du and F. Li are with the University of Tennessee, Knoxville, TN.

J. Li is with Old Dominion University, Norfolk, VA.

T. Zheng is with ISO New England, Holyoke, MA.

system, the input to the deep CNN will be the bus active power injection vector $\mathbf{P}_{inj}$, bus reactive power injection vector $\mathbf{Q}_{inj}$, bus conductance matrix $\mathbf{G}$, and bus susceptance matrix $\mathbf{B}$, and the output will be the $\mathbf{V}$ and $\boldsymbol{\theta}$ vectors. To further simplify the input data, we remove the $\mathbf{G}$ matrix and all the non-diagonal elements in the $\mathbf{B}$ matrix, and only keep the self-susceptance elements as the input. Since deep CNN regression is a data-driven method, the regression error caused by the missing data in $\mathbf{G}$ and $\mathbf{B}$ will be automatically made up via iterative training based on existing data samples. As such, the input to the deep CNN will be three $1 \times n$ vectors, $\mathbf{P}_{inj}$, $\mathbf{Q}_{inj}$ and $\mathbf{B}_{ii}$, and the output will be two $1 \times n$ vectors, $\mathbf{V}$ and $\boldsymbol{\theta}$. The simplification of the input data reduces both the model complexity of deep CNN and the computation burden, so it can be well applied to large-scale power systems.

C. Design of deep CNN structure

The structure of deep CNN for voltage angle calculation is illustrated in Fig. 1. It consists of two convolutional (Conv) layers and three fully-connected (FC) layer. The function of the convolutional layers is to extract features from the input power system raw data. Each convolutional layer is composed of a number of learnable convolution kernels, which are shown as purple squares in Fig. 1. A convolution kernel conducts the following convolution operation for feature extraction:

$$I_{new}(i, j) = \sum_{u=1}^c \sum_{v=1}^c I(u, v) \cdot \omega(u, v) + b \quad (3)$$

In (3), $I(u, v)$ is a single unit in the original input; $\omega(u, v)$ is a single unit in the convolution kernel square, which is also called the weight; c is the size of the convolution kernel; b is a bias parameter; $I_{new}(i, j)$ is a single unit in the output. As such, each unit in the output is the weighted sum of $c \times c$ units in the input, or we can say it contains the aggregated local information based on $c \times c$ input units. All the output units constitute a feature map. With more convolution kernels, more feature maps can be generated to provide sufficient information for deep CNN model regression.

In the constructed deep CNN, the convolution kernel size for the two layers are $[3, 3, 1, 12]$ and $[3, 3, 12, 24]$. The first two figures are the height and the width of the convolution kernel, the third figure is the depth of the kernel, and the last figure is the number of kernels. Zero padding is applied here to maintain the width and the height of the input. The generated feature maps further go through an activation function. The activation function will bring nonlinearity to the regression model (3). This is because the original mathematical relationship between bus power injection and bus voltage are not linear, as shown by (1)-(2), and cannot be fully represented by the linear convolution operation in (3). The limitation of linear transformations will be overcome by the activation function.

In this study, ReLU, or rectified linear unit, is applied as the activation function. The ReLU function has a quasi-linearity feature that makes it preserve high generalization ability as linear model, hence is applicable in deep CNN regression.

The output from the second convolutional layer will go through two separate fully-connected layers, FC1 and FC2. This is because in the designed deep CNN, two types of output will be generated. The first output is the system state variables,

i.e., the bus voltage angles and bus voltage magnitudes. The second output is the system security status. In this work, we adopt the security index defined in [5] to evaluate the N-1 contingency severity:

$$SI = \left[\sum_i \left(\frac{d_{v,i}^u}{g_{v,i}^u} \right)^{2n} + \sum_i \left(\frac{d_{v,i}^l}{g_{v,i}^l} \right)^{2n} + \sum_l \left(\frac{d_{p,l}^l}{g_{p,l}^l} \right)^{2n} \right]^{1/2n} \quad (4)$$

SI measures the margin between the system operation status and the system security boundary. More detailed explanations of (4) can be found in [5]. Based on the value of SI, the system security can be classified into three statuses: 1) $SI = 0$: secure; 2) $0 < SI \leq 1$: alarm; and 3) $SI > 1$: insecure.

In Fig. 1, the function of the fully-connected layers, FC1, FC2 and FC3, is to transform all the extracted features from the power system raw data to the desired output via matrix multiplication. Since security assessment is a classification problem, the *softmax* function is used as the activation function for the output from FC3, which differentiates various classes based on their posterior probabilities.

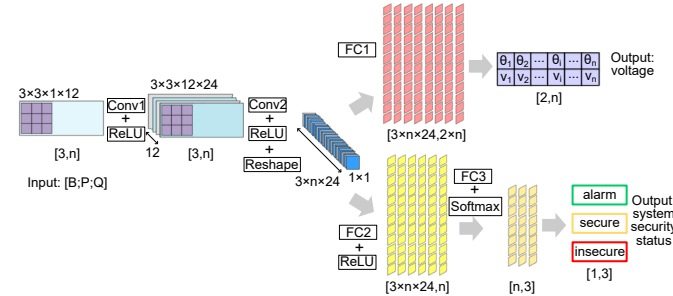


Fig. 1. Deep CNN structure for N-1 contingency screening using image-processing-like technique

It can be observed that the proposed deep CNN can realize two functions simultaneously: as an ACPF regression tool and as a classifier for system security assessment under N-1 contingency, which is a multi-task learning model. The loss function for training the deep CNN is defined as follows:

$$L = \frac{1}{N_s} \sum_{s=1}^{N_s} \left(\frac{1}{n} \left(\sum_{i=1}^n (\theta_{i,s}^* - \theta_{i,s})^2 + \sum_{i=1}^n (v_{i,s}^* - v_{i,s})^2 \right) - y_s^* \log(y_s) \right) \quad (5)$$

In (5), N_s is the number of training samples, and n is the number of buses. The first two terms are the mean square error of the bus voltage variables, where $\theta_{i,s}$ and $v_{i,s}$ are the deep CNN estimated voltage values, and $\theta_{i,s}^*$ and $v_{i,s}^*$ are the actual voltage values. The third term is called the cross-entropy, where y_s is the deep CNN estimated security classification, and y_s^* is the actual security classification. The cross-entropy is the most widely used loss function for multi-classification problems. Based on the above loss function, the deep CNN is trained via back-propagation method to decide the optimal network weight parameters and bias parameters.

III. CASE STUDY

The proposed image-processing-like, deep CNN model for ACPF calculation under N-1 contingency is tested on the IEEE 9, 30, 57, 118, and 300-bus systems, WECC 181-bus system, and European 1354-bus system to verify its accuracy and computing efficiency. To include multi-scenario uncertainty, Monte Carlo simulation is used to create load and renewable energy variations in training samples. For load uncertainty, we

assume that the real power of the bus load follows a uniform distribution within the range of [0.8, 1.2] of the base case, and the bus reactive load is calculated by multiplying the bus active power consumption with a factor uniformly drawn from the range [0.15, 0.25]. For renewable energy uncertainty, we change 40% of the conventional generators in the original test cases into wind generators, and the forecast error of wind generation follows a normal distribution with zero mean and a standard deviation of 0.05. For N-1 contingency, one line is randomly tripped in each training sample. The hardware environment for deep CNN training is a Nvidia GeForce GTX 1080 Ti Graphic Card with 11 GB memory and 1.582 GHz core clock. The software environment is the open-source deep learning platform TensorFlow for the proposed approach and MATPOWER for traditional model-based approach. The regression and classification results are shown in TABLE I.

In TABLE I, the errors of θ and v are the per unit mean absolute value over the test set compared with the results from model-based AC power flow calculation. The classification accuracy is the ratio between the number of test samples that has been correctly classified and the total number of test samples. As the table shows, deep CNN model possesses considerably high accuracy for ACPF calculation, even for large-scale power systems. Also, the training time is within an acceptable range given that the training is completed off-line.

TABLE I AC POWER FLOW RESULTS OF DEEP CNN

Case	No. of samples		Errors		Training time(s)	Classification Accuracy
	Training	Test	θ	v		
9	3292	1412	6.1e-3	7.2e-4	11.42	97.24%
30	4262	1066	1.5e-3	5.4e-4	23.06	96.25%
57	3360	1440	4.9e-3	1.6e-3	31.59	99.24%
118	3027	1298	7.5e-3	2.9e-4	57.88	100%
181(WECC)	2530	1085	5.7e-2	3.8e-3	65.04	97.70%
300	3445	1477	6.9e-2	2.3e-3	148.91	99.05%
1354 (Eu.)	3981	1707	1.1e-2	1.9e-3	1548.94	96.84%

To validate the computational efficiency of deep CNN regression, we compare calculation time of the AC power flow with N-1 contingency using both deep CNN and model-based AC power flow methods, as shown in TABLE II.

TABLE II TEST TIME COMPARISON

Case	Test size	Test time (s) (deep CNN)	Test time (s) (model-based)	Acceleration ratio
9	1412	0.017	3.500	206
30	1066	0.016	3.303	206
57	1440	0.018	4.323	240
118	1298	0.021	4.905	234
181 (WECC)	1085	0.025	4.655	186
300	1477	0.044	10.15	231
1354 (Eu.)	1707	0.264	34.13	129

In TABLE II, the last column shows the acceleration based on the computing time of deep CNN and the model-based ACPF method. The deep CNN approach is 129 to 240 times faster than the latter, average 205 times faster. This is because the well-trained deep CNN has high generalization to unseen test cases, and it can automatically generate AC power flow results and classify system security status under the new given input without any iterative calculation of power flows.

To further demonstrate the superiority of the proposed deep CNN over traditional artificial neural networks (ANNs), we design an ANN model with only one hidden layer. The size of the hidden layer is $[3 \times n, 3 \times n \times 24]$, which extracts the same

number of features as the deep CNN. Hence the two neural networks are comparable. The regression and classification results of ANN are shown in TABLE III.

TABLE III AC POWER FLOW RESULTS OF ANN

Case	No. of samples		Errors		Training time(s)	Test time (s)	Classification Accuracy
	Training	Test	θ	v			
9	3292	1412	2.0e-2	2.3e-3	6.52	0.011	91.64%
30	4262	1066	9.0e-3	2.4e-3	10.29	0.008	87.43%
57	3360	1440	2.7e-2	9.0e-3	14.88	0.009	92.43%
118	3027	1298	2.7e-2	9.0e-4	36.28	0.017	98.54%
181	2530	1085	1.8e-1	1.3e-2	47.31	0.019	75.94%
300	3445	1477	2.0e-1	5.7e-3	139.04	0.036	78.00%
1354	3981	1707	-	-	-	-	-

The results of the 1354-bus system are not available for ANN because the large-scale training data cause memory overflow. For all the other systems, deep CNN provides more accurate classification and regression results than the traditional shallow ANN. This is because the multiple convolutional layers within the deep CNN can extract better features for classification and regression, and this is the key contributing factor to the recent success of CNN in other applications. In addition, the traditional ANN is composed of fully-connected layers, where each neuron is connected to all of the subsequent neurons. This requires more neural parameters and computation. While in deep CNN, the sparse connectivity reduces both redundancy and computation to achieve better accuracy and computational efficiency. Note, although the deep CNN takes a little longer training time, it is of less importance since training is done offline.

IV. CONCLUSIONS

In this letter, a data-driven approach is proposed to achieve over 100 times acceleration for N-1 contingency screening under uncertain scenarios. An image-processing-like technique is proposed to utilize the deep CNN as an efficient regression method to fit AC power flow model, and then to automatically generate power flow results for system security evaluation with high accuracy. The high computational efficiency of deep CNN makes it a desirable tool for real-time security assessment applications as well as other related power system studies [1].

V. ACKNOWLEDGEMENT

The authors would like to thank the useful discussions with Eugene Litvinov, Feng Zhao, Jinye Zhao, Dane Schiro, and Izudin Lelic, all from ISO New England.

REFERENCES

- [1] F. Li and Y. Du, "From AlphaGo to Power System AI: What Engineers Can Learn from Solving the Most Complex Board Game," *IEEE Power & Energy Magazine*, vol. 16, issue 2, pp. 76-84, Mar. 2018.
- [2] M. Majidi-Qadikolai, R. Baldick, "Stochastic Transmission Capacity Expansion Planning with Special Scenario Selection for Integrating N-1 Contingency Analysis," *IEEE Trans Power Syst.*, vol. 31, pp. 4901-4912, Nov. 2016.
- [3] D. A. Tejada-Arango, P. Sánchez-Martín, and A. Ramos, "Security constrained unit commitment using line outage distribution factors," *IEEE Trans Power Syst.*, vol. 33, pp. 329-337, Jan. 2018.
- [4] I. Goodfellow, Y. Bengio, and A. Courville, "Deep Learning," Cambridge: MIT press, 2016, pp. 326-327.
- [5] R. Sunitha, R. S. Kumar, A. T. Mathew, "Online Static Security Assessment Module Using Artificial Neural Networks," *IEEE Trans Power Syst.*, vol. 28, no. 4, pp. 4328 - 4335, Nov. 2013.