Entropy Based Prioritization Strategy for Data-Driven Transient Stability Batch Assessment

Rong Yan, Zhaoyu Wang, Yuxuan Yuan, Guangchao Geng, and Quanyuan Jiang

***Abstract*—Transient stability batch assessment (TSBA) is essential for dynamic security check in both power system planning and day-ahead dispatch. It is also a necessary technique to generate sufficient training data for data-driven online transient stability assessment (TSA). However, most existing work suffers from various problems including high computational burden, low model adaptability, and low performance robustness. Therefore, it is still a significant challenge in modern power systems, with numerous scenarios (e.g., operating conditions and "N-k" contingencies) to be assessed at the same time. The purpose of this work is to construct a data-driven method to early terminate time-domain simulation (TDS) and dynamically schedule TSBA task queue a prior, in order to reduce computational burden without compromising accuracy. To achieve this goal, a time-adaptive cascaded convolutional neural networks (CNNs) model is developed to predict stability and early terminate TDS. Additionally, an entropy based prioritization strategy is designed to distinguish informative samples, dynamically schedule TSBA task queue and timely update model, for further simulation time reduction. Case study in IEEE 39-bus system validates the effectiveness of the proposed method.**

***Index Terms*—Cascaded convolutional neural networks (CNNs), dynamic task queue, entropy based prioritization strategy, time-domain simulation (TDS), transient stability batch assessment (TSBA).**

I. INTRODUCTION

TRANSIENT stability batch assessment (TSBA) gains popularity in power system planning and day-ahead dispatch, as numerous 'N-k' contingencies have to be assessed at this stage for dynamic security check [1]. It is also an imperative task and regular routine while accumulating training data for data-driven based transient stability assessment (TSA). Generally speaking, it consumes a huge amount of computational resources using current methods, like carrying out time-domain simulation (TDS) [2] and assessing stability according to rotor angle difference one-by-one. These engineering approaches lack scalability in solving large-scale problems. Therefore, it is necessary to develop a novel methodology for the TSBA task.

As known, TSA is a topic that has been extensively researched in the past few decades [3]. Normally, model-based methods are the mainstream of TSA. Among them, TDS [2] is the most reliable approach and acts as a standard to evaluate other methods, but suffers from the high computational burden. Moreover, TDS requires to solve a set of high-dimensional nonlinear differential algebraic equations (DAEs), and get the time-domain response of state variables. Unfortunately, there is no precise stability criterion for TSA. Only engineering experience-based stability criteria [3], [4] (the maximum rotor angle difference exceeding 180 degrees) is employed for TSA. Several other methods are developed to relieve the problem of consuming high computational resources, like transient energy function [5], extended equal area criteria [6], and trajectory convexity-concavity [7] method, but they are all facing the difficulties in model adaptability, and reliability [8]. Overall, similar to other nonlinear dynamic systems, the key challenge of the model-based TSA method in this task is still open.

Considering that the difficulties of model-based methods are not easy to break through in a short term, there has been a number of recent research on how to develop data-driven methods [9]. In general, the online computation cost of the data-driven methods is several orders of magnitude lower than that of the traditional model-based approaches. Basically, TSA task can be re-stated as a classification problem, which can be solved by various machine learning algorithms, like decision trees (DTs) [4], [10], [11], support vector machine (SVM) [12], [13], [14], local regression [15], convolutional neural network (CNN) [16]–[19], and recurrent neural network (RNN) [20]–[22]. Essentially, the basic idea of these solutions is to build a boundary in high-dimensional space to separate the data samples of different categories. The unique advantage of these data-driven model for online TSA is that the time-consuming training procedure can be carried out offline, thus enhancing scalability performance.

However, there are two main concerns for these data-driven methods. The first one is the lack of sufficient transient data samples. Considering that transient faults except one phase fault and reclosing are not common in daily operation of power grid, one possible solution is to obtain data by simulation. Thus, it is almost inevitable to consume huge amount of computational resources. The second one is the model adaptability. Model trained in one operation condition or topology is difficult to adapt to the scenario of others. As a result, more samples are required near the current operating conditions by tracing the time-variant operating conditions or grid topology. No matter what situations are, TSBA is a necessary task in this area with high requirements in both accuracy and efficiency performance.

As observed from the explanation mentioned above, TSBA

R. Yan is with the College of Electrical Engineering, Zhejiang University, Hangzhou, Zhejiang 310027, China and the Department of Electrical and Computer Engineering, Iowa State University, Ames, IA 50011, USA (e-mail: yanrong052@zju.edu.cn, yanrong@iastate.edu).

Z. Wang (Corresponding author) and Y. Yuan are with the Department of Electrical and Computer Engineering, Iowa State University, Ames, IA 50011, USA (e-mail: {wzy,yuanyx}@iastate.edu).

G. Geng and Q. Jiang are with the College of Electrical Engineering, Zhejiang University, Hangzhou, Zhejiang 310027, China (email: {ggc, jqy}@zju.edu.cn).

is a regular routine in both security check for gird planning and day-ahead dispatch, and data generation process of data-driven TSA. Unlike the online TSA, TSBA pursues overall high accuracy and efficiency performance in the batch rather than assessment response time for a single sample. According to this basic requirement, a cascaded CNNs model is designed in our previous work [1] specifically for TSBA task. Based on numerical results, our method outperforms the existing work in both accuracy and efficiency. Inspired by the concept of fine-tuning technique in medical image annotation [23], we select critical samples for priority assessment based on entropy, in order to dynamic updating assessment task queue and enhance model robustness as quickly as possible.

In this paper, we propose a data-driven transient stability batch assessment framework using cascaded convolutional neural networks (CNNs) and entropy based prioritization strategy. Noted that it is a continuation of the authors' previous work [1] in this area. This paper employ a cascaded CNNs model to determine the transient stability conclusion using simulated rotor angle waveforms and early terminate TDS. Moreover, in this work, an entropy based prioritization strategy is designed to distinguish informative samples, dynamically schedule TSBA task queue and timely update cascaded CNNs model, for further computational burden reduction. Overall, the proposed algorithm reduces the computational burden with more accurate assessment results, and improves the adaptability to time-variant operating conditions and grid topology.

II. TSBA Problem Formulation and Assumption

When the power system suffers from different disturbances under various operating conditions, rotor angles difference may change according to the severity of contingencies. Generally speaking, to obtain the time-series rotor angles data, a set of high-dimensional nonlinear differential algebraic equations are formulated using detailed model [24] of the investigated power systems as shown in Eq. (1).

$$\begin{cases} \dot{\boldsymbol{x}}(t) = \boldsymbol{F}(\boldsymbol{u}, \boldsymbol{x}(t), \boldsymbol{y}(t)) \\ \quad \boldsymbol{0} \; = \boldsymbol{G}(u, \boldsymbol{x}(t), \boldsymbol{y}(t)) \\ \boldsymbol{x}(0) = \boldsymbol{x}_0 \end{cases} \tag{1}$$

where $\boldsymbol{u}$ is steady-state operating point. $\boldsymbol{x}$ and $\boldsymbol{y}$ are respectively the time-variant state variables and operating variables that describe the dynamics of power grid in differential equations $\boldsymbol{F}$ and algebraic equations $\boldsymbol{G}$. $\boldsymbol{x}_0$ are the initial values of state variables. Then, the time solution of DAEs described in Eq. (1) can be obtained by numerical integration.

Based on the synchronizing characteristics of time-series rotor angles data aforementioned, all cases can be divided into three categories: stable, critical stable, and unstable. Actually, this classification process can be normally seen as the key to the TSA problem. As for a single TSA sample, one combination of specific operating conditions u and contingency k corresponds to a transient stability result. Moreover, the change of system configuration θ (e.g., grid topology) also greatly affects the stability conclusion. Thus, the mapping relation can be described as follows:

$$Stability\ Conclusion = f_{TSA}(u, k; \theta) \tag{2}$$

In the online TSA tasks, people pursue high accuracy and efficiency of a single assessment sample.

In practice, some tasks are quite different from online assessment, which is summarized in Table I. Large amount of scenarios are required to be assessed for "N-k" dynamic security check at the same time, especially in power system planning and day-ahead dispatch. Besides, a large amount of training samples are required for training procedures in data-driven based TSA methods. Therefore, we name this task as TSBA, to emphasize a large number of samples under different scenarios to be evaluated.

Prior to discussing the proposed algorithm, several assumptions or definitions are made to focus on TSBA task merely:

1) TSBA task set is pre-determined according to the specific requirements of application scenario, with detailed information of specific operating point and contingency. Note that the samples selection is out-of-scope for this paper, and can be referred to [8].

2) Samples with significantly different system configurations (e.g., grid topology) and operating points are recommended to be assessed in different batches. Different batches with similar system configurations and operating points are advised to be arranged adjacently in the TSBA task queue. Noted that it is not a limitation, and we just recommend doing so, in order to improve the overall efficiency performance. Detailed suggestions for batch division can be referred to Section IV-D of this paper.

3) As for contingency of each sample, they can be any possible large disturbance, like the contingency initiated by three-phase to ground fault(s) at any bus, and cleared after a random time by tripping a line connected to this bus.

In short, the key point is to find out the mapping relationship between partial TDS output and transient stability conclusion, dynamically arrange TSBA task queue, and timely update the mapping relationship, to save computational resources without compromising accuracy.

III. Time-adaptive TSA using Cascaded CNNs

A. Transient Stability Assessment using CNN Unit

1) Input Data Simulation and Pre-processing

In online TSA, PMU measurements are employed as the input of data-driven model. Different from online applications, the rotor angles can be easily obtained from simulator. As a result, the time-series rotor angles (time solution of DAEs described in Eq. (1)) output of time-domain simulator is assigned as the training data of CNN model, considering that the definition of transient stability is based on rotor angles. It should be noted that the output of simulator can not be directly fed to CNN, and data pre-processing is required. There are two steps [1]: one is re-sampling data with unequal time intervals using the interpolation method; and the other is eliminating numerical differences by normalization algorithm. In this way, multi-channel one-dimensional normalized data can be obtained and fed to CNN for subsequent TSA.

2) CNN Unit Structure for TSA

In this paper, CNN is employed to extract implicit features from time-series simulation output and to construct the

TABLE I
DIFFERENCES BETWEEN TSA AND TSBA

Task		TSA	TSBA
Sample for Assessment		One Given Sample	Thousands of Given Samples
System Configuration		One Given Configuration	One or Multiple Configurations
Operating Conditions (OC)		One Given OC	Multiple OCs within a Certain Range
Contingency		One Given Contingency	Multiple Possible Contingencies
Indices	Accuracy	Accurate Assessment Result	High Assessment Accurate Rate
	Efficiency	Low Assessment Latency	Low Average Assessment Time
Applications		Online Assessment	Offline Security Check, Training Data Generation

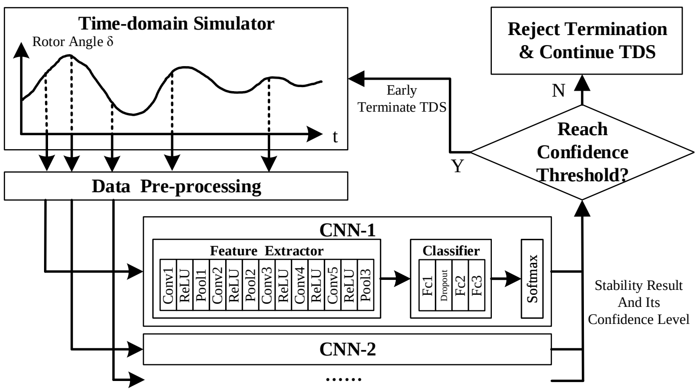


Fig. 1. Structure of time-adaptive cascaded CNNs.

mapping relationship between those extracted features and transient stability conclusion. Normally, a single CNN can be divided into two main parts, one is feature extractor, and the other is the classifier. The feature extractor part is composed of several convolution layers, pooling layers and activation layers, to extract features from normalized raw data in the way of encoding [25]. For the latter part, it is a multi-layer full-connected network. Besides, the dropout technique is employed to prevent over-fitting during training. Generally speaking, there is no universal rule for CNN unit design, and only some basic theory (like universal approximation theorem [26], [27]), experience, and "trial and error" [25] can be employed to design a model with superior performance. One possible architecture used in this paper is shown in a rectangular box marked with "CNN-1" in Fig. 1. As for data format in CNN unit, we treat the time-series simulation output as multi-layer-one-dimension data. After determining the structure of the model, the parameters of this model are trained by a widely-used method, stochastic gradient descent (SGD) algorithm with momentum [25]. It is a supervised learning method to minimize predefined loss function with cross-entropy loss and regularization loss.

B. Time-adaptive Cascaded CNNs Structure for TSA

Considering that the length of input data is fixed using a single CNN unit aforementioned, it is hard to determine a suitable time window of TDS for all kinds of cases. Normally, longer simulation time window results in higher reliability of assessment result. The confidence level of stability prediction result is still low if the time window of simulation is not long enough. Therefore, in this subsection, a time-adaptive cascaded CNNs model is introduced for TSA.

In order to handle the continuous time-series simulation output, several cutoff points are preset for simulator to output the time-series rotor angle data. Corresponding to each cutoff point, a CNN unit, discussed in the previous subsection, is employed to extract features from the simulation outputs, assess the stability conclusions, and then predict the confidence level. Several CNNs are then assembled and cascaded in a "relay" manner to realize such requirements. The output of each CNN unit is used as an index to measure the confidence level. Only if the confidence level reaches a preset stability threshold (like 0.999, 0.99), simulation can be terminated and stability conclusion can be determined accordingly. Otherwise, TDS continues until the next cutoff point for assessment. The detailed structure of time-adaptive cascaded CNNs model is shown in Fig. 1.

IV. ENTROPY BASED PRIORITIZATION STRATEGY FOR TSBA

A. Entropy Based Index for Priority Assessment

In our previous work [1], all samples to be assessed are treated equally, and as a result, the stability assessment process is carried out without considering the order of samples to be assessed. Although the previous work is easy to program, some implied rules which might be possible to further improve the efficiency can not be fully used.

Inspired by the concept of fine-tuning technique using entropy index in medical image annotation [23] and the least confidence index in [28], an entropy based prioritization strategy is proposed in this section. In this strategy, entropy index is employed to represent the average level of "information", "surprise", or "uncertainty" inherent in the data's possible outcomes [29]. Obviously, as for some samples, like obvious stable or unstable samples, the confidence level of stability assessment predicted by the cascaded CNNs model is close to 1 even if the model is not robust enough at the early stage of TSBA. As a result, we can conclude that there is little "information" inherited in these samples and they are useless for model enhancement. On the contrary, critical samples with confidence levels near 0.5 are much more informative and useful to construct the model. In other words, samples of this kind have a larger probability that are close to the security boundary and critical for model enhancement. Based on this idea, we are willing to find out the critical samples in the batch that have a significant effect on improving the performance of the cascaded CNNs model, at the beginning of TSBA.

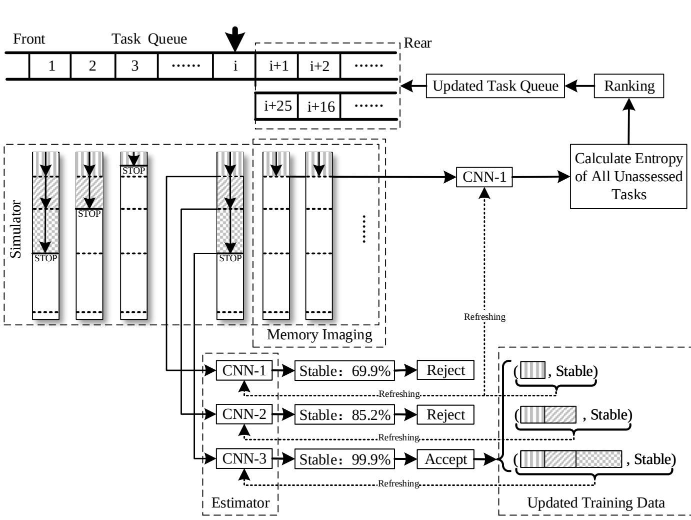


Fig. 2. Entropy based prioritization strategy for task queue update and model refreshing.

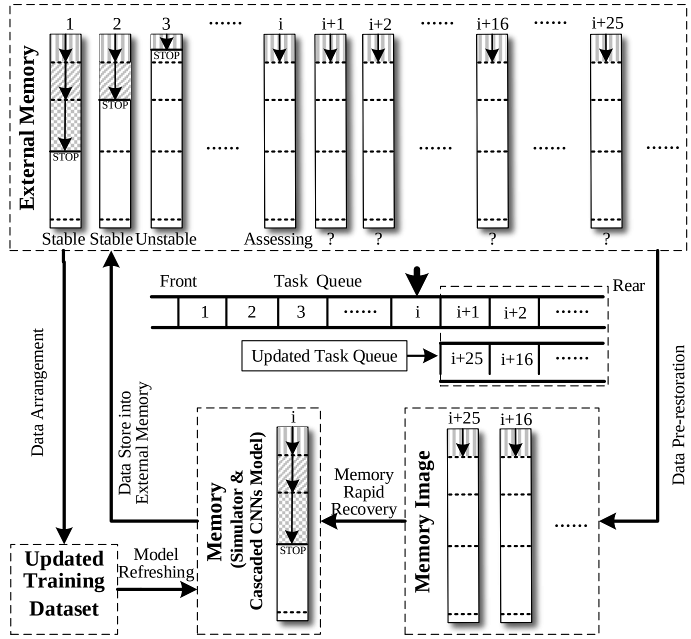


Fig. 3. Diagram of data storage flow using memory imaging technique.

In order to determine the worthiness of candidate samples for priority assessment, information entropy e_i is employed with definition of the i-th sample as follows:

$$e_i = -\sum_{j=1}^{L} \widetilde{y}_{ij} log \widetilde{y}_{ij} \tag{3}$$

where $[\widetilde{y}_{i1}, \widetilde{y}_{i2}, \widetilde{y}_{i3}]$ is the CNN outputs of the i-th sample, and three elements of this vector (the output of the cascaded CNNs model) represent the possibility that the sample is stable, critical stable, or unstable, respectively, according to the time-domain simulation of rotor angles. L is the number of stability categories. Generally speaking, sample with higher information entropy indicates that it is more useful to the model and should be assessed first. According to the value of entropy, the order of samples to be assessed in TSBA task queue can be updated. As a result, the performance of cascaded CNNs model can be improved more rapidly than our previous work, so as to shorten the simulation time window of the rest samples and improve the overall efficiency performance.

B. Entropy Based Prioritization Strategy for Task Queue Update and Model Refreshing

After defining an entropy based index that measures the quantity of information of a new sample, this index is leveraged to optimize the order of the TSBA task. For most samples, the simulation time window is larger than the first cutoff time (see sample 1, 2, and i in Fig. 2). Only the TDS time window of those obviously unstable samples, like sample 3 in Fig. 2, can be less than the first cutoff time because of the maximum rotor angle difference exceeding 180 degrees. Thus, the first CNN model is employed to predict the stability confidence level. As shown in Fig. 2, all samples simulate to the first cutoff point at the beginning, and calculate the value of entropy using Eq. (3). Based on this value, the order of all samples which have not been assessed can be updated. Note that samples with higher entropy are at the front of the task queue, in order to improve the accuracy and efficiency of the cascaded CNNs model as quickly as possible. In addition, each time the CNN-1 model is refreshed (explained below) while accumulating enough newly updated training data, the task queue would be updated again. For example, in Fig. 2, the CNN-1 is refreshed after assessing the i-th sample. At this point, the order of samples after the i-th sample is updated according to the lasted calculated entropy.

Moreover, the proposed cascaded CNNs model is required to be refreshed, in order to improve the performance of the model continuously. The samples which have been assessed can be fully utilized as the incremental training data. Take the i-th sample in Fig. 2 as an example, TDS can not be terminated until encountering the 3-rd cutoff point, since the confidence level predicted by the first two CNN units is not high enough. Therefore, we can obtain the time-series rotor angle data and the stability conclusion from CNN-3. It can be easily observed that stability conclusion predicted by a CNN-3 can be shared with CNN-1 and CNN-2 and employed to help to update these CNN units with the same or shorter length of simulation data. As the transient stability batch assessment continues, training data for the refreshing cascaded CNNs model accumulates. Once the number of updated training data reaches a preset number, the model will be refreshed accordingly. In such a way, the model will be more "experienced" in early terminate TDS and thus improve the performance of accuracy and efficiency.

C. Memory Imaging for Storage Acceleration

Different from the algorithm without considering the assessment order, in this paper, the proposed method requires to pause the process of TDS while encountering the cutoff points and to continue later. This seems to be the side effect of the proposed prioritization strategy because of the need of changing the order of samples to be assessed. To relieve this problem, the memory imaging technique is employed for storage and restoration acceleration. As shown in Fig. 3, external memory is used to store all simulation data and assessment conclusions. Before simulation, several samples are pre-restored from external memory to memory image. Once the former assessment task is finished, the data can be rapidly recovered from the memory image. Note that it does not matter whether carrying out TDS from scratch or going on simulation from the former cutoff point. After getting the stability conclusion using the cascaded CNNs model, data are stored into external memory again. On the contrary, there is no need to employ memory imaging technique to read data from external memory for model refreshing, considering that model refreshing can be carried out simultaneously with the process of assessment. Therefore, training data for model refreshing can be obtained from external memory directly.

D. Batch Division for Samples under Different Scenarios

In our previous work, all samples are assumed to be assessed at the same time in a single batch, and as a result, the performance is relatively poor if samples with significantly different scenarios are assessed at the same time. As known, for each CNN unit in this cascaded model, it matches a corresponding simulation time window. CNN unit with longer simulation time window is with better adaptability to changed operating points and topology, while the one with shorter time window is on the contrary. In other words, it is not sensitive to topology and operating points for the CNN unit corresponding to long simulation time window. For example, it is easy for CNN unit to get the stability conclusion after getting a 10 seconds' rotor angle waveform, regardless of topology and operating point of a given system. On the contrary, for CNN unit corresponding to short simulation time window, it is sensitive to these factors. In other words, the stability criterion of the first several CNN units is significantly different under the various scenarios, and obviously, samples with various scenarios may affect the training process of first several CNN units. As a result, it is difficult for cascaded CNNs model to obtain a reliable stability conclusion with high confidence level at the first several cutoff time, and the simulation time increases accordingly.

Therefore, a batch of samples is recommended to be separated into several sub-batches in order to improve the overall efficiency performance, according to the system configuration and operating conditions. Generally speaking, there is no strict restriction for us to decide whether the system configuration and operating conditions are significantly different, and only several suggestions are provided: 1) Large generator(s) scheduled to come online or offline; 2) Several critical transmission lines with heavy power flow scheduled to come online or offline at the same time; 3) Load increased or decreased larger than 40%. The above three cases can be seen as the criteria for batch division. Note that it is up to the user to decide whether to divide a batch of samples into several sub-batches in the end (although we recommend to do so) since it only affects the efficiency performance to some extent.

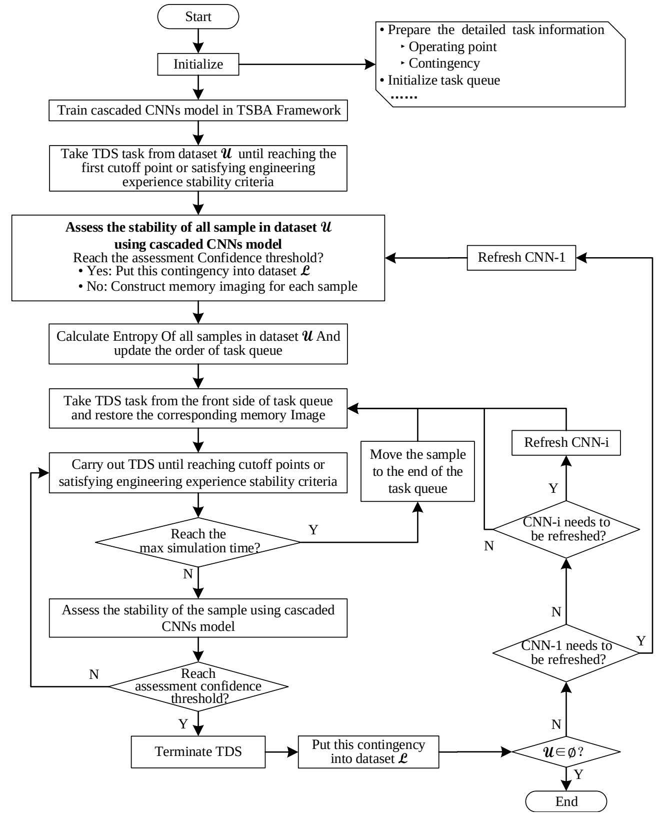


Fig. 4. Flowchart of the proposed TSBA algorithm.

E. TSBA Application Framework Using Entropy Based Prioritization Strategy

The flow chart of transient stability batch assessment is illustrated in Fig.4. In this figure, we define two separate datasets: labeled dataset ($\mathcal{L}$) and unlabeled one ($\mathcal{U}$), storing assessed samples and the ones to be assessed, respectively. In the beginning, all samples are stored in the $\mathcal{U}$, and $\mathcal{L}$ is empty. Samples stored in $\mathcal{U}$ are not unordered, and they are arranged in a predefined task queue which can be dynamically updated. It is worth noted that the initial order of the task queue is random.

Overall, the process of TSBA can be summarised as follows. The first step is to initialize the whole process of transient stability batch assessment. Specifically, we are required to prepare the detailed information (including operating points and contingencies for each sample) of batch assessment task, and to set up a task queue with random order in the dataset $\mathcal{U}$. It is worth noting that the TSBA task set is pre-determined, according to the user's specific requirement of application scenario. After clarifying the specific tasks of batch assessment, a small number of samples at the front of the task queue are forced to carry out TDS until reaching the end of the overall time window (normally 10 seconds) or satisfying engineering experience stability criteria (the maximum rotor angle difference exceeding 180 degrees), in order to obtain the stability conclusions. These samples are then moved into the dataset $\mathcal{L}$. In addition, CNN-1 (the model with the shortest cutoff time) is trained as a primitive model using those cases with stability conclusions. Noted that this step can be omitted if there exists a pre-trained CNN model using samples in similar operating conditions.

Once obtaining CNN-1, the output of this model can be regarded as a criterion for determining the worthiness of samples for priority assessment. In doing so, all cases in the $\mathcal{U}$ are required to simulate until the first cutoff time, and assessed by CNN-1 to capture the confidence level. Considering that lower confidence level value denotes higher degrees of information entropy, samples with relatively low classification certainty are selected to be assessed first in the next step, so as to improve the ability of cascaded CNNs model. Thus, the order of all samples in the task queue is updated according to this proposed index indicating classification confidence level.

So far, the simulator picks a TDS task from the front of task queue of dataset $\mathcal{U}$ and goes on simulation from the first cutoff time. Once encountering a cutoff point, it outputs time-series rotor angle data for cascaded CNNs model to extract features, predict stability probability. Only if the confidence level reaches the preset threshold, simulation can be terminated and this sample can be put into dataset $\mathcal{L}$. As the batch assessment continues, more updated training data are accumulated and thus they are employed to refresh the corresponding unit of the cascaded CNNs model. It is noted that the task queue in the dataset $\mathcal{U}$ is updated whenever CNN-1 is refreshed.

Follow the algorithm flow introduced above, all the samples in the task queue are assessed one-by-one until the samples in dataset $\mathcal{U}$ are empty. Meanwhile, a rubust cascaded CNNs model is constructed while batch assessment. For the same power grid operated in similar operating conditions, this trained model can also be used as the pre-trained CNN for further accelerate the process of batch assessment, to further simulation time reduction.

F. Performance Indices Evaluation

In order to evaluate the proposed algorithm and compare it with other existing methods, three types of indices (accuracy performance, efficiency performance, and memory consumption) are employed or defined in this subsection. Among them, all indices introduced in our former paper [1] are also employed in this paper. In this subsection, we just made a brief introduction. Moreover, the memory consumption of imaging each sample is added in this paper to measure extra space consumption of the proposed method.

1) Accuracy Performance: We define three indices to indicate accuracy performance. Among them, accuracy (AC) is defined as the percentage of samples which are correctly assessed. Besides, false dismissal (FD) and false alarm (FA) are to define two different statistical indices of misjudgment. FD defines the percentage of samples that misjudge unstable or critical stable samples as stable ones, and FA is just the opposite. What is noteworthy is that FD does far more harm to the power grid than FA. Therefore, we hope to find out an algorithm with minimum FD.

2) Efficiency Performance: To indicate the efficiency performance of proposed algorithm, average equivalent simulation time (AEST) is defined as Eq.(4), considering the impact of simulation time (ST), simulation time-consuming (STC), model training time (TT), and evaluation time (ET) on N samples.

$$AEST = \frac{1}{N}(\sum_{i=1}^{N} ST_i + \frac{\sum_{i=1}^{N} ST_i}{\sum_{i=1}^{N} STC_i}(\sum_{j=1}^{M} TT_j + \sum_{i=1}^{N} ET_i)) \tag{4}$$

In addition, marginal simulation time (MST) is defined to measure the potential of computational acceleration by Eq.(5).

$$MST = \frac{\sum_{i=N-500}^{N} ST_i}{500} \tag{5}$$

3) Memory Consumption: Three type of indices are employed to evaluate the requirement of space consumption. Specifically, model size provides the information of cascaded CNNs model complexity. Training data memory space consumption indicates the maximum memory space consumption of training data throughout the whole TSBA process. Simulator intermediate result memory consumption denotes the temporary memory space consumption of intermediate results of time-domain simulator.

V. Case Studies

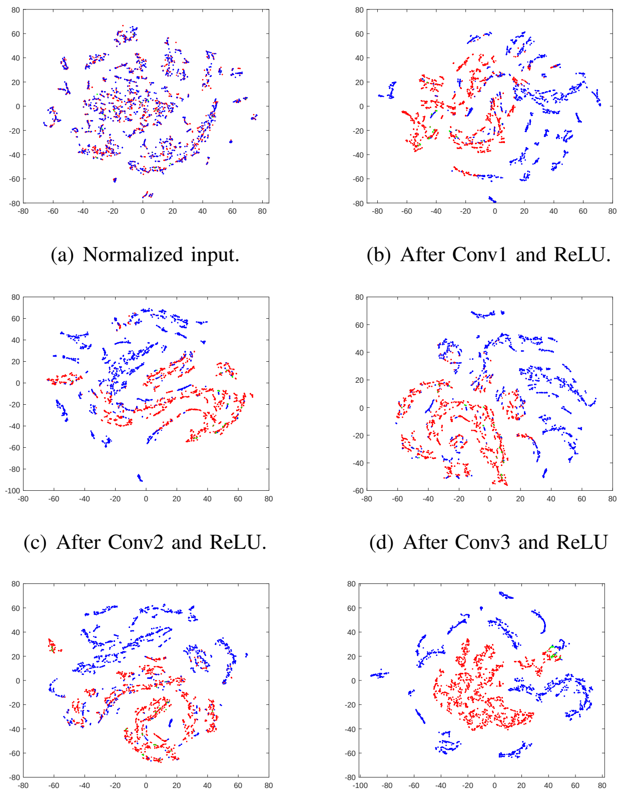
A. Test System and Configuration

In this section, the IEEE 39-bus test system is investigated to prove the effectiveness of the proposed method. The case study is done using three open-source packages: MATPOWER [30], PSAT [31], and TensorFlow [32]. Among them, the first one is employed to determine the steady-state operating conditions, the second one is used for time-domain transient stability simulation, and the last one is used for training CNNs. The case study is tested on a laptop with Intel Core i7-8850H 2.6GHz CPU, 16GB RAM.

To evaluate the proposed method, 10,000 samples are generated as the procedure introduced in section IV-A-1) of [1]. For data annotation, most samples are obviously stable and unstable according to the trend of rotor angles waveform and can be labelled directly. While, for other cases, like samples with persistently oscillated waveform, we prefer to label it as critical stable samples. But for more accurate, multiple authors in our lab are employed to label those samples independently, and the results can be determined according to the majority. More specifically, the dataset employed in this paper is the same as the one in [1], to ensure the reliability of the comparative test results.

B. CNN Structure Selection and Visualization for Better Feature Extraction

As we mentioned in the previous sections, the structure of a single CNN is illustrated. As known, more complex structure leads to higher accuracy of assessment results, but with longer assessment time. In other words, there is always a trade-off between accuracy and efficiency. However, there is no uniform network design method, and the design of detailed network, like structure and hyper-parameters, can only base on trial and error [25].



(a) Normalized input. (b) After Conv1 and ReLU. (c) After Conv2 and ReLU. (d) After Conv3 and ReLU (e) After Conv4 and ReLU (f) After Conv5 and ReLU

Fig. 5. 2-D Visualization using t-SNE (red points – stable contingencies, green points – critical stable contingencies, blue points – unstable contingencies).

For better explanation of the design process, the t-SNE [33] technique, which is regarded as the most popular visualized dimensionality reduction method, is employed to convert original features in high dimensions to representation space in 2 dimensions for visualization as shown in Fig 5. It is obvious that the points marked with different colors are distinguishable progressively, with the increase of the number of convolutional and activate layers. After 5 layers, the points with different colors can be clearly separated, although a small amount of points are still mixed together. Therefore, it is clear that 5 layers are enough for feature extraction.

C. Test Results

The initial CNN models are trained using 500 samples at the front of the task queue. Noted that the first 500 samples cannot be early terminated. Other samples are uniformly simulated to the first cut-off time except for the samples which can be terminated before that time according to the engineering experience stability criteria (the maximum rotor angle difference exceeding 180 degrees). All samples are then assessed using the first CNN model, and as a result, only 7,288 samples remain in the task queue for further investigation. Meanwhile, the samples in the task queue are reordered based on calculated entropy value. According to the order of the task queue, we assess each sample one-by-one. Once the samples used for updating model reach 500, the corresponding CNN is required to refresh the model using all samples available.

TABLE II
RESULTS COMPARISON USING IEEE 39-BUS SYSTEM

Method	AC(%)	FD(%)	FA(%)	RJ(%)	AEST(s)	MST(s)
EEM	98.02	1.98	0	0	5.175	4.909
TCC	93.46	2.28	0	4.26	2.632	2.633
SVM	99.20	0.57	0.23	0	4.660	≈ 0
DT	98.46	0.94	0.60	0	4.056	≈ 0
Single CNN	98.38	0.70	0.92	0	2.727	0.400
RNN	98.30	0.46	0.19	1.05	2.551	0.060
Cascaded CNNs	99.60	0.13	0.27	0	2.199	1.111
Proposed	99.72	0	0.28	0	2.044	0.808

1. The latest 500 contingencies employed to evaluate the potential of computational acceleration are different in the proposed methods compared with other approaches, as the original order of contingencies in the batch is changed according to the proposed algorithm. For the last 500 contingencies in the original order, the MST is 1.043s using the latest CNN models.
2. EEM: engineering experience stability criteria based on the results of time-domain simulation, details can be referred in part IV-3) of [4];
3. TCC: trajectory convexity-concavity stability criteria, which is proposed in [7];
4. Cascaded CNNs (cascaded convolutional neural networks), including algorithm framework, is developed in [1].
5. Other 4 methods are mainly modified from the existing online TSA frameworks. Generally speaking, more training samples result in higher accuracy and lower efficiency. However, for this batch assessment task, too many training samples increase the AEST. For comparison, we set the maximum false dismissal rate (1%) and error rate (2%) requirement , and the number of training samples can be determined accordingly: SVM (support vector machine): 9,000, DTs (decision trees): 8,000, single CNN (single convolutional neural network) and RNN (recurrent neural network): 5,000.

TABLE III
MEMORY CONSUMPTION TEST RESULTS COMPARISON

Method	Model Size	Training Data	Simulator Intermediate Results
EEM	≈ 0	≈ 0	≈ 0
TCC	≈ 0	≈ 0	≈ 0
SVM	0.99	142.21	≈ 0
DT	0.58	93.19	≈ 0
Single CNN	5.87	52.01	≈ 0
RNN	5.19	48.77	≈ 0
Cascaded CNNs	35.66	49.28	≈ 0
Proposed	35.66	30.98	11.40

* For the training data memory consumption of cascaded CNNs and the proposed model, the memory consumption for each CNN unit is different. The maximum training data memory consumption of those CNNs units are employed, considering that training data are usually stored in external memory unless the model is being trained.

In the whole assessment process for 10,000 samples, 6 CNN units are updated 8, 6, 4, 3, 2, and 1 times respectively, with the increase of model efficiency performance. Different from other CNN units, once the first CNN unit is updated, the task queue is refreshed accordingly, reflecting the latest sample information. Overall, to assess the whole 10,000 samples requires an average of 2.044s equivalent simulation time. As for the marginal time (the average of the last 500 samples to be assessed), the time window of simulation is only 0.808s.

Table II shows the comparison results of efficiency and accuracy between the proposed method and 7 other existing approaches. In terms of efficiency performance, TDS with engineering experience stability criteria is the most common method in real-world but suffers from long average simulation time. The proposed method reduces 60% average equivalent simulation time compared with this common method. Additionally, the proposed method also declines 22%-56% of average equivalent simulation time compared with trajectory convexity-concavity method and 4 other data-driven methods, respectively. Moreover, the proposed method also reduces

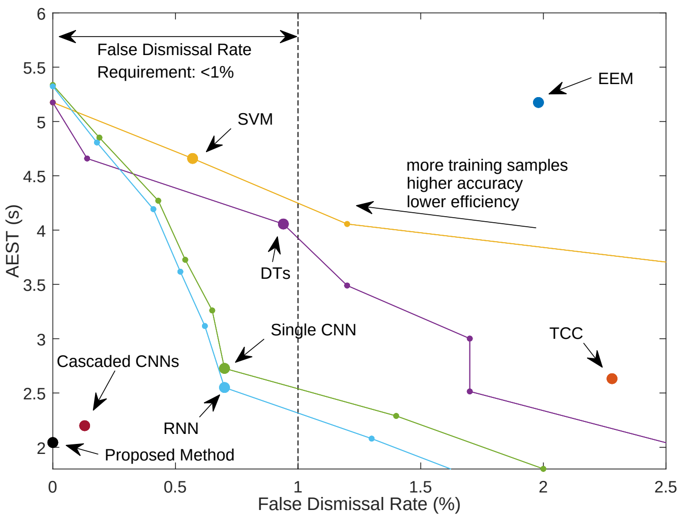
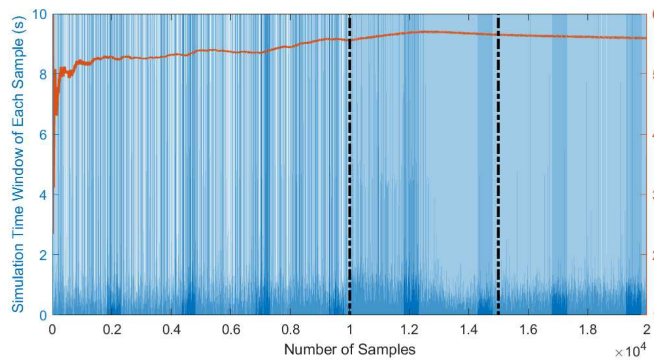


Fig. 6. The trade-off between accuracy and efficiency using various algorithms in IEEE-39 bus test system.

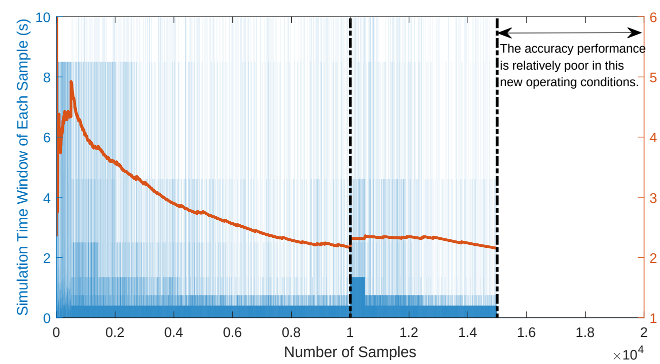
7.1% simulation time with our previous work, although the method shares similar structures with our previous paper. It is because critical samples are selected using the proposed entropy based prioritization strategy and assessed first, to get a model with superior performance at the early stage of batch assessment. As a result, the rest of samples in the batch can be assessed with a shorter simulation time window. In terms of accuracy performance, the results using the proposed algorithm is the best compared with other methods. In particular, the false dismissal rate is declined to 0, and it means that there is no misjudgment of unstable samples. It guarantees the credibility of the assessment results. In fact, entropy value of most false dismissal samples in our previous work is relatively high, and they are placed at the front of the task queue. As known, the performance of cascaded CNNs model is not robust enough to terminate time-domain simulation that early. In other words, the simulation time window of these samples are extended, and as a result, reducing the error rate. Note that, from the results, it seems that the proposed strategy is able to screen out critical samples that are easier to be misjudged, although it is not what we meant while designing this strategy.

Besides, the memory consumption comparison test result is illustrated in Table III. As shown in this table, memory consumption of model-based method (EEM and TCC) is approximately 0, since it only takes little memory for time-domain simulation of a single sample. On the contrary, data-driven based approaches are of small disadvantage in memory consumption. But it is obvious that the increased memory consumption can be ignored even for a common laptop. Compared with other data-driven methods, the model size and simulator intermediate result memory consumption of the proposed method is much larger, but it reduces the training data memory consumption at the same time. Overall, the proposed method outperforms both accuracy and efficiency performance, without compromising memory consumption.

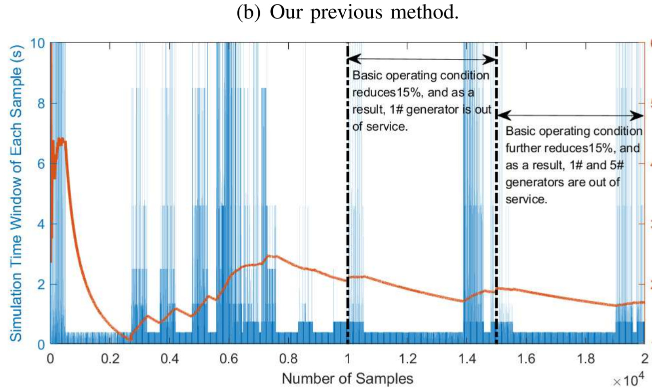
In fact, the key to transient stability batch assessment is to make a trade-off between accuracy and efficiency. Among them, what we need to avoid is to misjudge the unstable samples as stable ones, which would be harmful to power system. In Fig. 6, as observed that only the proposed method can reach the requirement of false dismissal rate of 0 with a relatively short simulation time window. Note that the AEST and error rate of those data-driven methods (SVM, RNN, single CNN, and DTs) modified from online applications differs with the change of the number of training data set in the batch. Therefore, we set a maximum false dismissal rate requirement to find out the best trade-off point for comparison.



(a) Engineering experience stability criteria.



(b) Our previous method.



(c) Proposed method in this paper.

Fig. 7. Robustness Test Results on Changed Operating Conditions and Rescheduled generators.

D. Robustness Test Results with Changed Operating Conditions and Rescheduled Generators

In order to verify the robustness of the proposed method, we assume that the operating condition baseline reduces 15% after assessing 10,000 samples and further reduces 15% after 15,000 samples. The generators are also rescheduled accordingly and two generators are out of service due to unit commitment. Different other data-driven methods lack adaptability, the proposed algorithm is trying to dynamically adjust the simulation time window, saving computation resources. As shown in Fig. 7(c), the model used for changed operating conditions does not need to train from scratch any longer, and the model used in the previous batch with similar operating conditions can be employed to initialize the model. As observed, the average simulation time window will increase slightly, but soon it decreases because of the timely model refreshing. For the whole 20,000 samples in three different scenarios, the average equivalent simulation time (the model training and prediction time is taken into account) is only 1.692 seconds. However, for our previous work, all samples are regarded as a single batch, although the operating points and topology have changed significantly. In other words, all samples under different scenarios in the batch are employed as the training data for cascaded CNN model refreshing after assessing 15,000 samples, no matter how different the samples are. Besides, our previous work is not able to find out the critical samples timely for model updating and thus affects both efficiency and accuracy performance. Overall, compared to the engineering experience method (see Fig. 7(a)) and our previous work (see Fig. 7(b)), the proposed method saves computational resources on the premise of adaptability to changed operating conditions and grid topology.

VI. Conclusions

This paper has proposed a novel data-driven transient stability batch assessment algorithm using entropy based prioritization strategy. In doing so, a time-adaptive cascaded CNNs model is employed to select a self-adaptive TDS time window for each sample to shorten the average simulation time without losses of accuracy. Additionally, an entropy based prioritization strategy is designed to dynamically schedule TSBA task queue and update the cascaded CNNs model. Overall, the proposed algorithm offers two advantages as follows:

1) Reducing the computational burden of TSBA task without losses of accuracy;
2) Improving the adaptability to time-variant operating conditions and grid topology;
3) Identifying informative samples quickly for priority assessment and model enhancement.

Future work will focus on the inherent limitation in current early termination methods, like cascading events may occur tens of seconds after large disturbance due to over current. We believe that a combination of data-driven based and probability based algorithms might be a possible solution for these hard problems. Furthermore, another potential future research topic is to develop a new TSA model, on the basis of this work, that is able to provide more information of which generator(s) is involved in the instability, or where is the "critical cluster", while obtaining the stability conclusions.

References

[1] R. Yan, G. Geng, Q. Jiang, and Y. Li, "Fast transient stability batch assessment using cascaded convolutional neural networks," *IEEE Trans. Power Syst.*, vol. 34, no. 4, pp. 2802–2813, Jul. 2019.

[2] M. Adibi, P. Hirsch, and J. Jordan, "Solution methods for transient and dynamic stability," *Proc. IEEE*, vol. 62, no. 7, pp. 951–958, Jul. 1974.

[3] P. Kundur, J. Paserba, V. Ajjarapu, and et al, "Definition and classification of power system stability," *IEEE Trans. Power Syst.*, vol. 19, no. 3, pp. 1387–1401, Aug. 2004.

[4] M. He, J. Zhang, and V. Vittal, "Robust online dynamic security assessment using adaptive ensemble decision-tree learning," *IEEE Trans. Power Syst.*, vol. 28, no. 4, pp. 4089–4098, Nov. 2013.

[5] H. Chang, C. Chu, and G. Cauley, "Direct stability analysis of electric power systems using energy functions: theory, applications, and perspective," *Proc. IEEE*, vol. 83, no. 11, pp. 1497–1529, Nov. 1995.

[6] Y. Xue, T. Huang, K. Li, Z. Dong, D. Yue, F. Xue, and J. Huang, "An efficient and robust case sorting algorithm for transient stability assessment," in *Proc. Power Energy Soc. General Meeting*, Jul. 2015, pp. 1–5.

[7] F. Su, S. Yang, H. Wang, and B. Zhang, "Study on fast termination algorithm of time-domain simulation for power system transient stability," *Proc. CSEE*, vol. 15, no. 15, pp. 4372–4378, Aug. 2017.

[8] R. Yan, G. Geng, and Q. Jiang, "Data-driven transient stability boundary generation for online security monitoring," Apr. 2020. [Online]. Available: https://arxiv.org/abs/2004.01369

[9] L. Duchesne, E. Karangelos, and L. Wehenkel, "Recent developments in machine learning for energy systems reliability management," *Proc. IEEE*, vol. 108, no. 9, pp. 1656–1676, Sep. 2020.

[10] I. Genc, R. Diao, V. Vittal, S. Kolluri, and S. Mandal, "Decision tree-based preventive and corrective control applications for dynamic security enhancement in power systems," *IEEE Trans. Power Syst.*, vol. 25, no. 3, pp. 1611–1619, Aug. 2010.

[11] C. Liu, K. Sun, Z. H. Rather, Z. Chen, C. L. Bak, P. Thgersen, and P. Lund, "A systematic approach for dynamic security assessment and the corresponding preventive control scheme based on decision trees," *IEEE Trans. Power Syst.*, vol. 29, no. 2, pp. 717–730, Mar. 2014.

[12] F. Gomez, A. Rajapakse, U. Annakkage, and I. Fernando, "Support vector machine-based algorithm for post-fault transient stability status prediction using synchronized measurements," *IEEE Trans. Power Syst.*, vol. 26, no. 3, pp. 1474–1483, Aug. 2011.

[13] Y. Dai, L. Chen, W. Zhang, and Y. Min, "Multi-support vector machine power system transient stability assessment based on relief algorithm," in *2015 Asia-Pacific Pow. Energy Eng. Conf., APPEEC*, Nov. 2015, pp. 1–5.

[14] J. Geeganage, U. Annakkage, T. Weekes, and B. Archer, "Application of energy-based power system features for dynamic security assessment," *IEEE Trans. Power Syst.*, vol. 30, no. 4, pp. 1957–1965, Jul. 2015.

[15] S. Li, V. Ajjarapu, and M. Djukanovic, "Adaptive online monitoring of voltage stability margin via local regression," *IEEE Trans. Power Syst.*, vol. 33, no. 1, pp. 701–713, Jan 2018.

[16] Z. Shi, W. Yao, L. Zeng, J. Wen, J. Fang, X. Ai, and J. Wen, "Convolutional neural network-based power system transient stability assessment and instability mode prediction," *Appl. Energy*, vol. 263, p. 114586, Feb. 2020.

[17] B. Claessens, P. Vrancx, and F. Ruelens, "Convolutional neural networks for automatic state-time feature extraction in reinforcement learning applied to residential load control," *IEEE Trans. Smart Grid*, vol. 9, no. 4, pp. 3259–3269, Jul. 2018.

[18] Y. Chen, Y. Wang, D. Kirschen, and B. Zhang, "Model-free renewable scenario generation using generative adversarial networks," *IEEE Trans. Power Syst.*, vol. 33, no. 3, pp. 3265–3275, May. 2018.

[19] H. Wang, H. Yi, J. Peng, G. Wang, Y. Liu, H. Jiang, and W. Liu, "Deterministic and probabilistic forecasting of photovoltaic power based on deep convolutional neural network," *Energy Convers. Manage.*, vol. 153, pp. 409–422, Dec. 2017.

[20] Y. Zhou and Y. Yang, "Transient stability assessment using gated recurrent unit," in *Proc. Asia Conf. Power Electr. Eng., ACPEE*, Jun. 2020, pp. 560–564.

[21] J. Yu, D. Hill, A. Lam, J. Gu, and V. Li, "Intelligent time-adaptive transient stability assessment system," *IEEE Trans. Power Syst.*, vol. 33, no. 1, pp. 1049–1058, Jan. 2018.

[22] F. Pan, J. Li, B. Tan, C. Zeng, X. Jiang, L. Liu, and J. Yang, "Stacked-gru based power system transient stability assessment method," *Algorithms*, vol. 11, no. 8, p. 121, Aug. 2018.

[23] Z. Zhou, J. Shin, L. Zhang, S. Gurudu, M. Gotway, and J. Liang, "Fine-tuning convolutional neural networks for biomedical image analysis: Actively and incrementally," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit, CVPR*, Jul. 2017, pp. 4761–4772.
[24] P. Kundur, *Power System Stability and Control*. New York, USA: McGraw-Hill, 2006.
[25] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. MIT Press, 2016.
[26] K. Hornik, "Approximation capabilities of multilayer feedforward networks," *Neural networks*, vol. 4, no. 2, pp. 251–257, 1991.
[27] B. C. Csáji, "Approximation with artificial neural networks," *Faculty of Sciences, Etvs Lornd University, Hungary*, 2001.
[28] Y. Zhou, Q. Guo, H. Sun, Z. Yu, J. Wu, and L. Hao, "A novel data-driven approach for transient stability prediction of power systems considering the operational variability," *International Journal of Electrical Power Energy Systems*, vol. 107, pp. 379 – 394, May 2019.
[29] Wikipedia. (2020, Aug.) Entropy (information theory). [Online]. Available: https://en.wikipedia.org/wiki/Entropy_(information_theory)
[30] Z. Ray and M. Carlos. (2018, Mar.) Matpower: a matlab power system simulation package. [Online]. Available: http://www.pserc.cornell.edu/matpower/
[31] F. Milano. (2018, Mar.) Power system analysis toolbox (psat). [Online]. Available: http://faraday1.ucd.ie/index.html
[32] Google. (2018, Mar.) Tensorflow. [Online]. Available: http://www.tensorflow.org/
[33] L. Maaten and G. Hinton, "Visualizing data using t-sne," *J. Mach. Learn. Res.*, vol. 9, no. Nov., pp. 2579–2605, 2008.

Rong Yan received his B.S. degrees in electrical engineering from the School of Electrical Engineering, Wuhan University, Wuhan, China, in 2016. Currently, he is pursuing the Ph.D. degree at the College of Electrical Engineering, Zhejiang University, Hangzhou, China. From 2019 to 2020, he was a visiting student at the Department of Electrical and Computer Engineering, Iowa State University, Ames, United States.

His research interest is the application of data-driven methods in power system stability analysis.

Zhaoyu Wang received the B.S. and M.S. degrees in electrical engineering from Shanghai Jiaotong University in 2009 and 2012, respectively, and the M.S. and Ph.D. degrees in electrical and computer engineering from the Georgia Institute of Technology in 2012 and 2015, respectively. He is the Harpole-Pentair Assistant Professor with Iowa State University. His research interests include power distribution systems and microgrids, particularly on their data analytics and optimization. He was the recipient of the 2020 IEEE Power and Energy Society (PES) Outstanding Young Engineer Award. He is the Secretary of IEEE PES Award Subcommittee, a Co-Vice Chair of PES Distribution System Operation and Planning Subcommittee, and a Vice Chair of PES Task Force on Advances in Natural Disaster Mitigation Methods. He is an Associate Editor of IEEE Transactions on Power Systems, IEEE Transactions on Smart Grid, IEEE Power and Energy Society Letters, IEEE Open Access Journal of Power and Energy, and IET Smart Grid.

Yuxuan Yuan received the B.S. degree in electrical and computer engineering from Iowa State University, Ames, IA, USA, in 2017, where he is currently working toward the Ph.D. degree.

His research interests include distribution system state estimation, synthetic networks, data analytics, and machine learning.

Guangchao Geng received his B.S. and Ph.D. degrees in electrical engineering from the College of Electrical Engineering, Zhejiang University, Hangzhou, China, in 2009 and 2014, respectively. From 2012 to 2013, he was a visiting student at the Department of Electrical and Computer Engineering, Iowa State University, Ames, United States. From 2014 to 2017, he is a post-doctoral fellow at the College of Control Science and Engineering, Zhejiang University, Hangzhou, China and the Department of Electrical and Computer Engineering, University of Alberta, Edmonton, AB, Canada.

Currently, he is an Associate Professor at the College of Electrical Engineering, Zhejiang University, Hangzhou, China. His research interest includes power system stability and control, the applications of internet of things (IoT) technique and high performance computing (HPC) technique in power systems.

Quanyuan Jiang received his B.S., M.S., and Ph.D. degrees in electrical engineering from Huazhong University of Science & Technology, Wuhan, China in 1997, 2000, and 2003, respectively. From 2006 to 2008, he was a visiting associate professor at the School of Electrical and Computer Engineering, Cornell University, Ithaca, United States.

He is currently a professor at the College of Electrical Engineering and the academic dean of Graduate School, Zhejiang University, Hangzhou, China. His research interest includes power system stability and control, applications of energy storage systems and high performance computing technique in power systems.