IEEE Open Access Journal of
Power and Energy

Received 8 December 2023; revised 7 March 2024; accepted 17 April 2024.
Date of publication 29 April 2024; date of current version 8 May 2024.

Digital Object Identifier 10.1109/OAJPE.2024.3395177

Predictive Online Transient Stability Assessment for Enhancing Efficiency

RUI MA, SARA EFTEKHARNEJAD (Senior Member, IEEE), AND CHEN ZHONG

Department of Electrical Engineering and Computer Science, Syracuse University, Syracuse, NY 13210 USA
CORRESPONDING AUTHOR: S. EFTEKHARNEJAD (seftekha@syr.edu)

This work was supported in part by the National Science Foundation (NSF) under Grant 2144918.

ABSTRACT Online transient stability assessment (TSA) is essential for the reliable operation of power systems. The increasing deployment of phasor measurement units (PMUs) across power systems provides a wealth of fast, accurate, and detailed transient data, offering significant opportunities to enhance online TSA. Unlike conventional data-driven methods that require large volumes of transient PMU data for accurate TSA, this paper develops a new TSA method that requires significantly less data. This data reduction is enabled by generative and adversarial networks (GAN), which predict voltage time-series data following a transient event, thereby minimizing the need for extensive data. A classifier embedded in the generative network deploys the predicted data to determine the stability of the system. The developed method preserves the temporal correlations in the multivariate time series data. Hence, compared to the state-of-the-art methods, it is more accurate using only one sample of the measured PMU data and has a shorter response time.

INDEX TERMS Classification, generative adversarial networks, phasor measurement unit, transient stability.

NOMENCLATURE

δ_i	Rotor angle of machine *i*.
ω_i	Rotor speed of machine *i*.
σ	Sigmoid activation function.
B_{ij}, G_{ij}	Conductance and susceptance in the *i*th row and *j*th column of admittance matrix $\mathbf{Y}$.
D, G	Discriminator and Generator of the GAN model.
D_i	Damping constant of generator *i*.
e (e_h)	Prediction error at each (h_{th}) forecast time step.
E_i	Internal bus voltage of machine *i*.
h_{t-1}	Output from the previous GRU.
h_t	Final GRU output.
$h_t^{'}$	The current memory unit of GRU.
M_i	Angular momentum of machine *i*.
P_i^m, P_i^e	Mechanical/electrical power input of *i*th machine.
r_t	The reset gate output of GRU.
$tanh$	Hyperbolic tangent activation function.
x_t	Input data to GRU.
z_t	The update gate output of GRU.

I. INTRODUCTION

THE increased levels of power grid uncertainties has renewed calls for rapid online transient stability assessment (TSA) tools that will grant the system operators adequate time for mitigative actions, such as intentional islanding or controlled load shedding. Conventional TSA methods belong to one of three categories: 1) time-domain simulations, 2) direct methods, and 3) trajectory-based approaches. Time-domain methods solve a set of high-dimensional and nonlinear differential-algebraic equations to assess transient stability [1], [2]. These methods assume complete knowledge of the system model parameters and its operating conditions. However, their high computational complexity makes them impractical for near real-time applications. To enable a faster TSA, direct methods have been developed, e.g., the Lyapunov method [3], [4], [5] and the extended equal area criterion [6], [7]. The direct methods simplify TSA by energy functions that evaluate the dynamic performance of the system [8], [9]. Yet, these methods do not scale well because of the need to simplify the dynamic models. Trajectory-based methods, such as Lyapunov exponents [10] and apparent impedance [11] methods, are relatively fast. However, at least a few cycles of post-fault transient data are needed to warrant TSA accuracy.

With the increased availability of high-resolution Phasor Measurement Unit (PMU) data, new TSA approaches have emerged where physical system constraints and system data inform each other for TSA. Hence, without a complete dynamic grid model, the stability status of the power grids can be determined. Machine learning (ML) methods such as decision trees [11], [12], [13], core vector machine [14], support vector machine (SVM) [15], and extreme learning machine [16], [17] have been deployed for TSA using system measurements, formulating TSA as a two-class (stable and unstable) classification problem. These methods transform raw PMU data into specific features before feeding them into the model. This transformation process can inadvertently lead to the loss of critical dynamic information inherent in the raw data, potentially compromising the accuracy of TSA. Deep learning methods, such as the convolutional neural network [18], [19], long-short-term memory (LSTM) [20], [21], gated recurrent unit (GRU) [22], and stacked denoising autoencoder [23] have addressed information loss by directly utilizing raw PMU data. However, to guarantee assessment accuracy, longer durations of post-contingency time series data are needed, which will result in undesirable TSA delays. Given the critical importance of rapidly assessing transient events in power systems, there is a significant need for shorter TSA response time.

To address the aforementioned gaps in data-driven TSA, a new and accurate TSA approach is developed that requires shorter durations of post-contingency measurements, thereby significantly reducing the total response time. The novelty of the developed TSA lies in its ability to *predict* the post-contingency measurements that are used for classification. The prediction capability is enabled by extending Generative Adversarial Networks (GAN), a model-free ML method for generating datasets that closely mimic real transient datasets.

A GAN model consists of two neural networks: generative and adversarial networks. The generative network converts the input data, drawn from a Gaussian distribution, to synthetic data. The adversarial network classifies the generated synthetic data as either real or synthetic. The objective of the adversarial network is to maximize the difference between real and synthetic data, while the generative model aims to minimize this difference [24]. GAN has been deployed in various domains for generating synthetic datasets, including images [25] and music [26]. In power system applications, GAN is deployed to generate missing PMU data [27] or address the lack of realistic PMU time series data [28], [29], [30]. Unlike the existing work that aimed to enhance the historical data, in this paper, GAN is extended for *predicting* the future PMU data. Hence, a different approach to adopting GAN is required to generate realistic future data and predict the system transient state based on those data. This approach should: a) learn the temporal features of the transient data; b) predict the post-fault PMU data; and c) assess the transient stability of the system. The developed GAN-based TSA approach addresses these requirements for a fast and accurate stability assessment.

The novelty of the developed TSA approach stems from the ability to rapidly and accurately determine transient stability while only utilizing one sample of the measured post-contingency PMU data. This is achieved by developing a hierarchical structure, shown in Fig. 1, for the refined GANs. Several GAN models are stacked to construct the HGAN. The hierarchical generative adversarial network (HGAN) input is the measured post-contingency PMU data. The lowest level GAN utilizes the measured PMU data to predict the measurements for the next sampling time. Higher-level GANs use predictions to further predict the PMU time series data. With the proposed structure, HGAN requires only a single sample of the measured data to predict the post-contingency time series transient data. In addition to the predicted sequence of the transient data, a binary classifier is embedded in the generative network of each sub-GAN. Each sub-GAN performs TSA individually, based on the measured one sample PMU data and the predicted subsequent time-series data from the lower-level sub-GANs. The TSA is achieved by combining the assessment results from all sub-GANs with the aim of reducing the error. As raw measured data are used to determine the transient status in the developed method, the possibility of losing critical dynamic information during feature selection is avoided. The main contributions of this work are as follows:

1) A hierarchical GAN-based TSA is developed to predict the transient status after a disturbance. Only a single sample of post-contingency PMU data is needed. Hence, TSA can be performed in a short time. Here, as the voltage enables providing insights into the ability of the system to maintain stability, PMU voltage data is utilized;
2) The generative and adversarial networks are designed to retain the temporal features of the multivariate PMU time-series data and improve TSA accuracy;
3) The case studies demonstrate that in comparison to the conventional ML-based TSA, the developed model achieves better performance in terms of accuracy and speed.

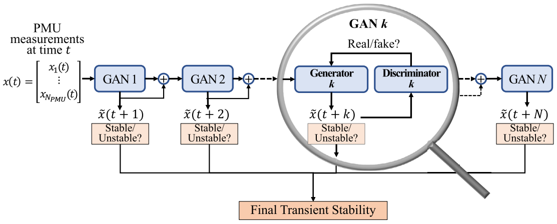


FIGURE 1. **The structure of the developed HGAN-based TSA.**

This paper is organized as follows: Section II formulates the TSA problem. Section III introduces the conventional GAN model and illustrates each element of the developed

TSA. Section IV tests the developed HGAN model on the IEEE 118-bus system. Conclusions are provided in Section V.

II. TRANSIENT STABILITY ASSESSMENT

During a power system disturbance, such as a transmission line fault, large swings appear in generator rotor angles, bus voltages, or line currents [23]. To ensure a stable grid, all generators should maintain synchronism after the clearance of the faults. If synchronism is lost, prompt mitigations such as controlled islanding need to be taken to prevent further failures. This remarks on the need for an online TSA approach that can promptly and precisely identify the transient state at the earliest stages of a fault clearance.

In a system with n generators and loads modeled as constant impedances, the equation of motion for generator i is [31],

$$\dot{\omega_i} = \frac{P_i^m - P_i^e - D_i\omega_i}{M_i} \tag{1.1}$$

$$\dot{\delta_i} = \omega_i \tag{1.2}$$

$$P_i^e = E_i^2 G_{ii} + \sum_{j\neq i}^{n} [E_iE_jB_{ij}sin(\delta_{ij}) + E_iE_jG_{ij}cos(\delta_{ij})] \tag{1.3}$$

$$\delta_{ij} = \delta_i - \delta_j. \tag{1.4}$$

The rotor speed, electrical and mechanical powers are functions of the rotor angle. Since the rotor angle is typically used to determine the transient stability, analyzing these three terms, i.e., P^m, P^e, and ω, can also indirectly infer the system transient stability. During a large disturbance, electrical power will change immediately. Variations in mechanical power and rotor speed, due to inertial and governor response, occur after the first swing, when stability is assessed. The difference in the response time leads to a net accelerating power and may cause an unstable transient. As the imbalanced power is mainly due to sudden changes in electrical power P^e, the change in electrical power can be used to assess the transient stability of the system. Furthermore, as observed in (1.3) that the electrical power P^e is a function of the voltage, the trajectory of the voltage can thus be used to study the transient.

During a transient event (fault), large excursions occur in the trajectories of the generator rotor angles and bus voltages. To classify the transient stability status of the system, a stability index based on rotor angles is commonly used [14], [20], [32]:

$$\eta = \frac{360^\circ - \Delta\sigma_{max}}{360^\circ + \Delta\sigma_{max}} \tag{2}$$

where $\Delta\sigma_{max}$ is the maximum post-fault angle difference between two generators. The stability is defined as,

$$\phi = \begin{cases} \text{stable;} & \eta > 0 \\ \text{unstable;} & \eta \leq 0. \end{cases} \tag{3}$$

It may take tens of cycles to obtain $\Delta\sigma_{max}$, which is not desired for online TSA. The developed TSA acquires the transient status in the early stages of a contingency. Unlike the other ML-based TSA methods that require a large volume of post-contingency PMU data, the developed method only requires the first sample of the PMU data after fault clearance.

Here, the PMU voltage data are selected for TSA. As mentioned in [33], the generator rotor angle cannot be measured directly by PMUs. Since the focus here is on determining the transient stability of the system from PMU data, voltage magnitudes are used. Various data-driven TSA methods have used voltage data to study the transients [20], [33], [34], [35], [36]. The studies in [35] and [36] demonstrate that TSA algorithms with voltage magnitudes are fast and eliminate the errors and extra pre-processing needed to calculate an angle reference. As explained in Section IV-B only utilizing PMU voltage magnitude data is sufficient for an accurate TSA. However, other measurements that contain the response of the system can also be used by the developed TSA method. Only some buses are equipped with PMUs. PMUs are added in a manner to ensure full system observability (discussed in Section IV-A).

III. PREDICTIVE ONLINE TSA

To address the gap in the existing data-driven TSA techniques, a new approach is developed for near real-time, which is accurate and requires significantly less PMU data than the conventional methods. The developed method takes a twofold approach: 1) predicting the post-contingency PMU voltage time-series data and 2) TSA using the predicted data. In this section, as the prediction capability of the developed TSA approach is enabled by GAN and gated recurrent units (GRUs), fundamentals of the GAN and GRUs are introduced first. Then, a detailed description of each element of the developed HGAN-based TSA is discussed.

A. THE GENERATIVE ADVERSARIAL NETWORK

GAN is an unsupervised learning approach that has been used in many applications for generating synthetic images, music, time series data, among others [25], [26], [37], [38], [39]. GAN learns the distribution of a real dataset and generates synthetic data that are close to the real data [24]. In the context of power systems, GAN-based methods have been developed to recover missing PMU data [27] and generate synthetic PMU datasets [28], [29].

GAN consists of two deep neural networks: generator G and discriminator D. The generator generates synthetic data while retaining the statistical features of the real data. The discriminator differentiates the real data from the fictitious data generated by the generator. The two networks G and D are trained iteratively until a Nash equilibrium is achieved, which indicates that the generated synthetic data and real data cannot be distinguished [27].

Specifically, for a random Gaussian noise space $\mathbf{Z}$, the generator maps the input noise z drawn from $\mathbf{Z}$ to a synthetic dataset $\hat{x}$ as,

$$\mathbf{G}(z|\theta_g) : z \rightarrow \hat{x} \tag{4}$$

Algorithm 1 The GAN training steps

```
Initialize the neural network parameters of the generator and discriminator, i.e., $\theta_g$ and $\theta_d$ respectively;
for $ep = 1, \cdots, ep_{max}$ do
    Obtain a minibatch of the random noise $z$ from $P(z)$, and another minibatch of the real data $x$ from $P(x)$;
    Generate the synthetic data $\hat{x} = G(z|\theta_g)$;
    Feed the generated synthetic data $\hat{x}$ and the real dataset $x$ into the discriminator $D$, and obtain the estimated probability $D(x|\theta_d)$ and $D(\hat{x}|\theta_d)$;
    Update $G$ and $D$ by descending their gradients:
        $\nabla_{\theta_g}(\mathbb{E}_{z\sim P(z)}[log(1 - D(\hat{x}))])$
        $\nabla_{\theta_d}(\mathbb{E}_{x\sim P(x)}[logD(x)] + \mathbb{E}_{z\sim P(z)}[log(1 - D(\hat{x})])$
end
```

where θ_g is the neural network parameters of G. The objective of the generator is to map a random noise to synthetic data, such that the distribution of the synthetic data $P(\hat{x})$ is close to the real data $P(x)$. In other words, the objective of G is to minimize the difference between the real and synthetic data,

$$\underset{G}{\text{Min}}\ V(G, D) = \mathbb{E}_{z\sim P(z)}[\log(1 - D(\hat{x}))]. \tag{5}$$

The discriminator D estimates the probability, $D(\cdot|\theta_d)$, that the input data is real rather than synthetic, where θ_d the is the neural network parameters of D. The discriminator aims to maximize the difference between the real and synthetic data so that they could be distinguished,

$$\underset{D}{\text{Max}}\ V(G, D) = \mathbb{E}_{x\sim P(x)}[\log D(x)] + \mathbb{E}_{z\sim P(z)}[\log(1 - D(\hat{x})] \tag{6}$$

where $V(\cdot)$ is the loss function. The first term $\mathbb{E}_{x\sim P(x)}[\log D(x)]$ is the probability that the discriminator classifies the real data as real. The second term is the probability that the synthetic data $\hat{x}$ generated from G is classified as fake by D.

As the generator and discriminator are trained together, the objective of GAN is to address a minimax problem:

$$\underset{G}{\text{Min}}\underset{D}{\text{Max}}\ V(G, D) = \mathbb{E}_{x\sim P(x)}[\log D(x)] + \mathbb{E}_{z\sim P(z)}[\log(1 - D(\hat{x}))]. \tag{7}$$

The GAN training algorithm is described in Algorithm 1. In each training episode, by leveraging the stochastic gradient descent algorithm, the generator and discriminator are updated, such that the synthetic data are closer to the real data.

B. GATED RECURRENT UNITS FOR FEATURE EXTRACTION

When predicting the future dynamic response in a power system, it is critical to preserve the temporal features of the time series data. GRUs, a variant of RNN, enable this by learning the temporal features of time series data [40]. RNN retains information from past data and combines it with present data to accurately predict the future sequence of data. If RNN is not designed properly, the gradient may exponentially update towards zero, referred to as a vanishing gradient, or exponentially diverge to infinity, referred to as exploding gradients. With a mechanism to determine the volume of past data, GRU eliminates the problem of vanishing or exploding gradients. As illustrated in Fig. 2, GRU consists of three elements: the reset gate, the update gate, and the current memory unit [40]. The update gate determines the volume of past data used to predict the future data sequence. The reset gate determines the volume of the stored past data to be forgotten. The current memory unit uses the output from the reset gate to store the relevant information from the past data. The GRU output is the sum of the output from the three gates. The three gates and the final output can be described as,

$$r_t = \sigma(W^r x_t + U^t h_{t-1}) \tag{8.1}$$

$$z_t = \sigma(W^z x_t + U^z h_{t-1}) \tag{8.2}$$

$$h'_t = \tanh(Wx_t + r_t \odot Uh_{t-1}) \tag{8.3}$$

$$h_t = z_t \odot h_{t-1} + (1 - z_t) \odot h'_t. \tag{8.4}$$

In (8), W and U are the weights of the GRU network, and $\odot$ is the Hadamard product (elementwise). This GRU structure predicts the transient sequence with only one sample of the measured PMU data. This critical advantage is enabled by learning the temporal features of the transient data. The added discriminator to the GRU, enables further capturing the distribution of the transient data.

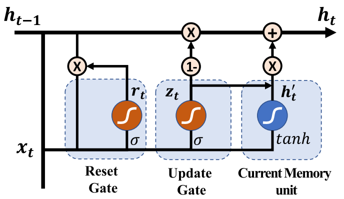


FIGURE 2. The structure of GRU.

C. THE DEVELOPED HGAN-BASED TSA METHOD

The HGAN-based TSA, shown in Fig. 1, predicts post-fault transients and the system's transient stability status. The GAN structure is improved by learning the temporal features of the multivariate PMU time-series data. As the deployment of one type of PMU data is sufficient to assess the transient stability, PMU voltage magnitude data are utilized for TSA. However, other PMU measurements could also be used. The developed HGAN adjusts the conventional GAN model to predict the PMU sequence data. The conventional GAN takes random noise to produce synthetic data, while the refined GAN takes real measurements as input to predict the transient sequence.

Here, the measured post-fault PMU voltage magnitudes are the inputs to the GAN model, i.e., $x(t)$. Data for the next time, that is, $\hat{x}(t + 1)$, are predicted by the aforementioned

GAN structure. However, the measured data $x(t)$ and the predicted data $\hat{x}(t+1)$ may not have sufficient information to determine the transient status. To address this deficiency, a multi-step prediction strategy, particularly a hybrid direct recursive strategy, is utilized to predict the transient sequence and uncover its oscillation patterns to address this deficiency. The hybrid direct-recursive model uses predictions from previous time steps to forecast values in future time steps, incorporating both direct and recursive strategies. Assume h is the prediction horizon. The direct strategy uses observed PMU data to predict voltage magnitude $\hat{x}_{t+h}$, as $\hat{x}_{t+h} = f_h^{\text{direct}}(x(t)) + E_h^{\text{direct}}$, where $f(\cdot)$ is the prediction model and E_h is the accumulated prediction error at the h_{th} time step. In the direct strategy, separate models are developed for each time step. Since only one sample of PMU data is available and separate models are used, direct strategies cannot learn the temporal correlations between the predictions. As a result, E_h^{direct} is relatively large. In the recursive strategy (9), the one-step-ahead prediction model uses the previous predicted data as input.

$$\hat{x}_{t+h} = \begin{cases} f^{\text{recursive}}(x(t)) + E_h^{\text{recursive}} & h = 1 \\ f^{\text{recursive}}(\hat{x}(t+h-1)) + E_h^{\text{recursive}} & h > 1 \end{cases} \tag{9}$$

Since the correlations between the sequence data are retained, the prediction error is smaller than the one using direct strategy, i.e., $E_h^{\text{recursive}} \leq E_h^{\text{direct}}$. However, as the prediction model is used recursively at each time step, the prediction error $E_h^{\text{recursive}} \approx \sum_1^h e$ will accumulate as h increases. To overcome the shortcomings of the direct and recursive strategies, a hybrid direct-recursive strategy (10) is developed [41],

$$\hat{x}_{t+h} = \begin{cases} f_h^{\text{hybrid}}(x(t)) + E_h^{\text{hybrid}} & h = 1 \\ f_h^{\text{hybrid}}(\hat{x}(t+h-1), \cdots, \hat{x}(t+1), x(t)) & h > 1 \\ \qquad + E_h^{\text{hybrid}} & \end{cases} \tag{10}$$

In this hybrid model, a separate prediction model is trained for each forecast time step, where the model uses the previously predicted and observed PMU data as input. The accumulated prediction error at time step h can be represented as $E_h^{\text{hybrid}} \approx \sum_1^h e_h$. Since separate models are trained for each time step, the prediction error is smaller than the recursive strategy, i.e., $e_h \leq e$. Hence, the hybrid strategy achieves a smaller accumulated error, that is, $E_h^{\text{hybrid}} \leq E_h^{\text{recursive}}$. In short, the multi-step prediction using hybrid direct-recursive strategy enables generating accurate and longer sequence data than the direct and recursive strategies. Therefore, more distinct transient oscillation patterns are uncovered for accurate TSA.

In this study, the hybrid direct-recursive strategy is to stack multiple GANs to predict a longer sequence of future data. With N stacked GANs, the measured post-fault voltage data are fed into the lowest level GAN, i.e., *GAN 1* to predict the data for the next time instant $\hat{x}(t+1)$. The predicted data and the measured data $x(t)$ and fed into *GAN 2* for subsequent predictions. Therefore, the HGAN-based TSA with N GANs generates $N+1$ voltage time series data, $\{x(t), \hat{x}(t+1), \cdots, \hat{x}(t+N)\}$.

A binary classifier is embedded in each generator of the stacked GANs for TSA. In *GAN k*, the input sequence data $\{x(t), \hat{x}(t+1), \cdots, \hat{x}(t+k-1)\}$ is deployed to classify stability as stable or unstable. The classification results from all sub-GANs are combined by an average-based ensemble strategy to determine the final transient stability assessment. The generator has three objectives: 1) learn the distribution of the real measurements; 2) predict the next sequence of the time series data; 3) assess the stability status. Based on these objectives, the *GAN k* loss function is formulated as,

$$\begin{aligned} \mathbf{L}_{G_k} &= \log(1 - D(\hat{x}(t+k))) + (\hat{x}(t+k) - x(t+k))^2 \\ &\quad + y\log(p(y)) + (1-y)\log(1-p(y)) \end{aligned} \tag{11}$$

where y is a binary label, with $y = 0$ assigned to unstable and $y = 1$ to stable transients. The probability of a stable classification is denoted by $p(y)$. In the loss function in (11), the square of the error, $(\hat{x}(t+k) - x(t+k))^2$, is used to minimize the error of the predicted data $\hat{x}(t+k)$. The cross entropy loss $y\log(p(y)) + (1-y)\log(1-p(y))$ is included so that the generator correctly classifies the system stability status. The structure of the discriminator D is consistent with the conventional GAN, where the difference between the real and predicted data will be maximized. The GRU structure is used to construct adversarial and generative networks to better learn the temporal features of the time series data.

The structure of *GAN k* ($k \in \{0, 1, \cdots, N\}$) is illustrated in Fig. 3. The input to *GAN k* is the measured and predicted PMU voltage data $X = \{x(t), \hat{x}(t+1), \cdots, \hat{x}(t+k-1)\}$, $X \in \mathbf{R}^{k \times N_{PMU}}$, where N_{PMU} is the total number of PMUs. The GRU cell in the generator learns the temporal features of the time series data from the input X. The fully-connected layer is added to the GRU to improve learning efficiency. As the generator has two different outputs, i.e., the predicted label and subsequent data, two parallel layers are used to perform the two tasks of classification and prediction separately. To generate the predicted label y and the predicted data $\hat{x}(t+k)$, the activation functions *softMax* and *Sigmoid* are used to interpret the output of the fully-connected layer. The output from the discriminator GRU is sent to the fully-connected layer, followed by a linear activation function that converts the output to the estimated probability of the input data $D(\cdot|\theta_d)$. The probability is backpropagated to the generator to update the parameters of the generative network. Maximizing the discriminator loss function during training (reformulated in (12)), leads to maximal $D_{\text{optimal}}(x) = \frac{P(x)}{P(x)+P_g(x)}$ [24], where $P_g(\cdot)$ is the distribution of the data generated from G and $P(\cdot)$ is the real data distribution. The optimal probability D_{optimal} minimizes the generator, so the distributions of the real and predicted transient data are the same, i.e., $P(\cdot) = P_g(\cdot)$. Hence, with the learning signal from discriminator, the transient data can be better learned in generator. Detailed proof for the optimal discriminator and generator can be found in [24].

$$V(G, D) = \int_x p(x) \log D(x) dx + \int_z p(z) \log(1 - D(g(z))) dz \\ = \int_x p(x) \log D(x) + p_g(x) \log(1 - D(x)) dx \quad (12)$$

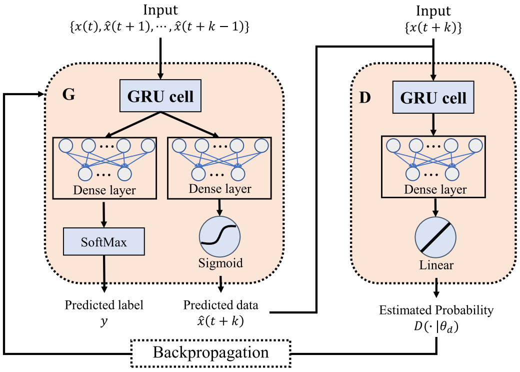


FIGURE 3. **The structure of GAN *k* in the HGAN-based TSA.**

The prediction accuracy of the upper level GAN depends on the accuracy of the lower GANs. A small prediction error from a lower GAN propagates to higher levels and eventually lead to a large error. We address this problem by sequentially training GANs, i.e., when training *GAN k*, the network parameters of the trained lower GANs i, $(i \in \{0, \cdots, k-1\})$, are kept fixed. Otherwise, repeatedly updating the parameters of the lower GANs will lead to overfitting.

To improve the learning efficiency of the HGAN-based TSA, the training data is normalized first. Here, the min-max normalization is used so that the maximum normalized voltage is one. Starting from the lowest level GAN, the sub-GANs will be trained sequentially. For each GAN model, by deploying the minibatch stochastic gradient descent algorithm, the generator and the discriminator are updated iteratively until the convergence of the cross-entropy loss. The well-trained HGAN-based TSA can be deployed for online TSA. Upon measuring the first post-fault PMU voltage data, $v(0)$, the HGAN-based model with N sub-GANs generates the subsequent voltage data $\{v(0), \hat{v}(1), \cdots, \hat{v}(N)\}$. The TSA results of these N sub-GANs will be combined using an average-based ensemble technique for the final TSA classification. Here, the final TSA result is assumed to be stable if more than half of the sub-GANs vote for stable; otherwise, the system is unstable. This averaging process takes into account the unique attributes of each sub-GAN. Therefore, it diminishes the impact of individual model errors, and effectively reduces the overall prediction variance [42].

IV. CASE STUDIES

The IEEE 118-bus system is studied under various conditions. The performance of HGAN-based TSA is compared with decision trees, SVM, LSTM, GRU, and stacked GRU. As the TSA accuracy of the HGAN-based method is based on PMU measurements, sensitivity studies are performed to investigate the impact of noise, the number and the location of PMUs on the TSA accuracy. Simulations are conducted on an i7 computer with a 3.2GHz CPU and 64GB RAM.

TABLE 1. **PMU placement in the IEEE 118-bus system.**

Test System	PMU placement
IEEE 118-bus system	5, 12, 15, 17, 32, 37, 49, 56, 59, 67, 69, 70, 71, 77, 80, 85, 92, 96, 100, 105

A. PMU DATA GENERATION

The post-fault PMU voltage time series for testing and training are generated with simulations using the Transient Security Assessment Toolbox (TSAT) of DSATools [32]. Here, as demonstrated in subsection IV-B, it is found that only PMU voltage magnitude data is sufficient for accurate TSA.

Various operating conditions, ranging from 90% to 110% of the nominal load, are simulated under normal conditions and with one line out of service (due to maintenance). Note that this study assumes load variations are confined to a single season. To accommodate the substantial variations across different seasons and significant grid upgrades, it is suggested to train the GAN-based TSA model separately for each specific season. Furthermore, in scenarios where there are substantial upgrades in grid topology, updating and retraining the GAN-based TSA model is essential to maintain the prediction accuracy. The transient scenarios considered are three-phase faults on every bus and transmission line (located at 30%, 50%, and 80% of the lines). In alignment with other TSA studies, faults in our simulations are cleared after 5 cycles [35]. Additionally, breaker auto-reclosure is simulated to occur five cycles after the initial fault clearance. Note that simulation data are collected after the clearance of faults. To minimize the adverse impacts of an imbalanced dataset on the accuracy of TSA, 5000 stable and 5000 unstable transients are randomly selected from the generated transients. Furthermore, 80% of the data are used for training and the remaining 20% for testing. As the stability index (2) is widely used in TSA studies [14], [20] and industry, this study utilized this index to create the ground truth information.

To generate the PMU data, the simulation sampling frequency is set at 120 frames per second. To mimic real power grids where not every bus is equipped with PMU, 20 PMUs, determined by the minimum spanning tree method [43], [44], [45], are deployed in the IEEE 118-bus system to ensure full observability. The locations of the PMUs are provided in Table 1 and can be found in Fig. 4. The training and testing data can be found at the IEEE DataPort (https://dx.doi.org/10.21227/6f5v-q924). Note that this study compares the proposed method with other state-of-the-art methods using the same dataset.

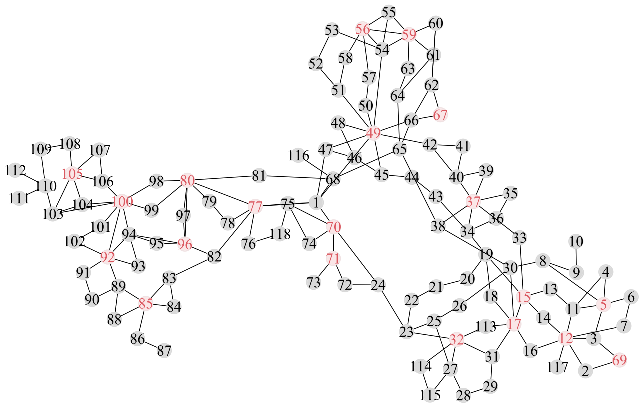


FIGURE 4. IEEE-118 bus system.

TABLE 2. Parameters of the HGAN-based TSA model.

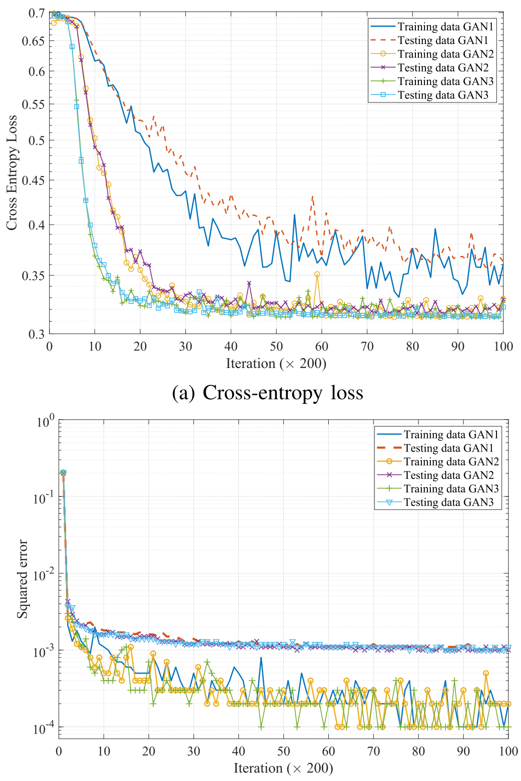
Parameters	Value
Number of GANs	3
Number of episode	20000
Learning rate of generator / discriminator	1e-3 / 1e-4
Number of GRU layers / hidden layers	2 / 30
Batch size	128

B. PERFORMANCE ANALYSIS OF THE HGAN-BASED TSA

—Performance Analysis: Upon obtaining the datasets for transient events, each level GAN is trained. The training parameters are provided in Table 2. The HGAN-based model with three GANs is found to be sufficient for an accurate TSA. Although more GANs generate a longer duration of voltage time-series data and better represent a transient pattern, a small error in lower-level GANs propagates through the stacked GANs and eventually leads to a large cumulative error. The learning rates of the generator and discriminator are set to 0.001 and 0.0001, respectively. A larger rate increases the learning speed, but the results may be suboptimal.

To stabilize training and avoid GAN mode collapse, as suggested by [37], the generator is trained twice, and the discriminator is trained once in each training episode. The training time for three sub-GANs is approximately 3.5 hours with CPU. As the training of the HGAN is only conducted once and offline and can be significantly reduced with GPU, this training time is reasonable.

To demonstrate the performance of the HGAN-based model, the cross-entropy loss and the squared error of each sub-GAN are depicted in Fig. 5. The test data (not used for training) check the model for overfitting. As seen in Fig. 5, the cross-entropy loss and the squared error dramatically decrease at the beginning of the training. Furthermore, with predicted data from lower GANs, the cross-entropy loss of *GAN*2 and *GAN*3 drops faster than the loss in *GAN*1, indicating that the combination of predicted and measured data leads to more distinguishable transient features for the GAN. Since the cross-entropy loss of the testing data does not increase as the loss of the training data decreases, no overfitting occurs.



(a) Cross-entropy loss

(b) Squared error

FIGURE 5. The cross-entropy loss and the squared error for each of the three GANs in the HGAN-based model.

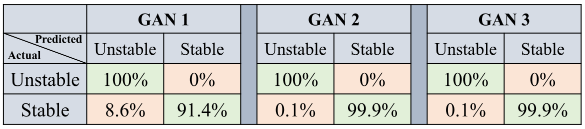
Upon training the HGAN model, the testing data, i.e., 1000 stable and 1000 unstable events, are used to evaluate the classification accuracy, which is presented in Table 3. With the single measured PMU voltage data, the classification accuracy of *GAN*1 is 95.35%. However, combining the real measured data and the predicted data from the lower level GANs yields an accuracy of 99.95% for *GAN*2 and *GAN*3. Since a single sample of data contains limited transient information, the classification accuracy of *GAN*1 is low. With more predicted data that contain the temporal features of a transient event, the hidden oscillation patterns of the event are more apparent to the classifiers and result in higher classification accuracy.

The confusion matrix for each GAN is shown in Fig. 6. For *GAN*1, all unstable transient data are correctly identified, while 8.6% of the stable events are incorrectly classified. Benefiting from the predicted data, in *GAN*2, the percentage of misclassified stable events drops to 0.1%. Combining the classification results of these three GANs with an average-based ensemble strategy yields a TSA accuracy of 99.95% under various conditions. The computation time for TSA with three GANs is 0.00298s (0.359 cycles). As GAN1, GAN2, and GAN3 share an identical structural framework,

TABLE 3. TSA classification accuracy (F1 score) of each GAN in the HGAN-based TSA.

	GAN 1	GAN 2	GAN 3
Accuracy	95.88 %	99.95 %	99.95 %

the execution time of each GAN is equal. The total response time, including the waiting time for one cycle of the PMU data, is 1.359 cycles, which is reasonable for near real-time.



	GAN 1		GAN 2		GAN 3	
Predicted Actual	Unstable	Stable	Unstable	Stable	Unstable	Stable
Unstable	100%	0%	100%	0%	100%	0%
Stable	8.6%	91.4%	0.1%	99.9%	0.1%	99.9%

FIGURE 6. The confusion matrix of each GAN.

—Comparison with Baseline Methods: The developed TSA is compared to conventional data-driven methods that deploy decision trees, SVM, LSTM, and GRU. The inputs to the baseline methods are the measured post-fault PMU voltage data. For the highest classification accuracy, the LSTM hyperparameters, namely, the learning rate, training iteration, the number of layers, hidden layers, and batch size, are set to 0.0005, 40000, 2, 40, and 128, respectively. For GRU, the optimal learning rate, training iteration, number of layers, and hidden layers are 0.0005, 40000, 2, and 50, respectively. Comparisons are presented in Table 4 and Fig. 7. Compared to SVM, LSTM, and GRU, the decision trees and the HGAN-based TSA are more accurate, using only one sample of PMU measurements. The HGAN-based method reaches an accuracy greater than 99% in a shorter time. The SVM, GRU, and LSTM-based TSA achieve similar classification accuracy at around two cycles. The reduction of 0.65 cycles achieved by the developed method is critical for online TSA. The response time is the sum of the time for PMU samples to be available and the TSA computation time. Although decision tree TSA achieves a high classification accuracy with one data sample, unlike the developed HGAN-based TSA, decision trees do not predict the transients. The predictive capability makes the developed method suitable for further applications, such as stability margin prediction and cascading failure analysis. Furthermore, even though the training time for HGAN is much longer compared to decision trees, the training is carried out offline. Hence, its real-time utility is not affected.

TABLE 4. TSA accuracy of the baseline methods and the HGAN-based TSA with only one sample of the 20 PMU measurements.

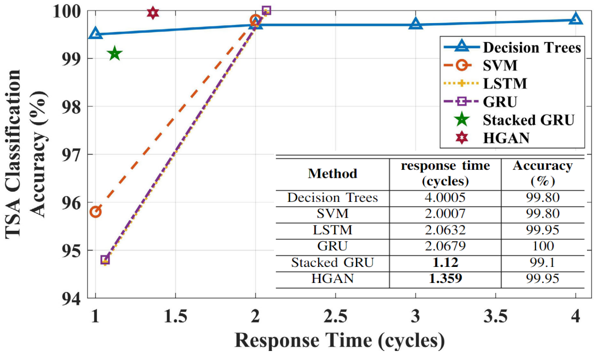
Method	TSA Classification Accuracy
Decision tree	**99.5%**
SVM	95.8%
LSTM	94.75%
GRU	94.8%
Stacked GRU	**99.1%**
HGAN	**99.95%**

TABLE 5. TSA classification accuracy (F1 score) of each GRU in the stacked-GRU model.

	GRU 1	GRU 2	GRU 3
Accuracy	95.1 %	98.2 %	98.7 %

A stacked-GRU model is used for TSA to demonstrate the benefits of the discriminator in the HGAN model, where only the generator is retained. The stacked-GRU has the same configuration as the HGAN model. The classification accuracy, response time, and the accuracy of each sub-GRU are given in Table 4, Fig. 7 and Table 5, respectively. The stacked-GRU also learns the temporal correlations of the sequence data and achieves a high TSA classification accuracy, i.e., 99.1% in 1.12 cycles, as opposed to 99.95% in 1.359 cycles with the HGAN-based method. Compared to the TSA classification accuracy of each sub-GAN in Table 3, $GAN2$ and $GAN3$ achieve higher accuracy than $GRU2$ and $GRU3$ in Table 5. The discriminator helps the generator better capture the dynamics of the transients, hence resulting in higher accuracy. Since the HGAN-based model and the stacked-GRU have a similar response time and the TSA accuracy is more critical, the HGAN-based model outperforms the stacked-GRU.

In summary, HGAN-based TSA outperforms the other baseline methods in terms of accuracy and applicability. Although in [22] a stacked-GRU model was developed to classify transient events, this model is a multilayer GRU, where the higher-level GRU uses the output of the lower-level GRU in the same time step. This stacked-GRU model is the baseline GRU model studied in Table 4, with two GRU layers. Furthermore, this model can only be used for classification and requires at least several measurement cycles to achieve high TSA accuracy.



Method	response time (cycles)	Accuracy (%)
Decision Trees	4.0005	99.80
SVM	2.0007	99.80
LSTM	2.0632	99.95
GRU	2.0679	100
Stacked GRU	**1.12**	99.1
HGAN	**1.359**	99.95

FIGURE 7. Comparison of response times for IEEE 118-bus system (20 PMUs). To achieve the same accuracy as the HGAN TSA, other methods use more data and have a longer response time.

C. IMPACT OF PMU MEASUREMENT VARIATIONS

The simulation-generated PMU measurements are noise-free. In practice, PMU measurements are contaminated by

TABLE 6. **PMU placement scenarios.**

Scenario	Location
1	5, 12, 15, 17, 37, 49, 59, 69, 71, 77, 80, 85, 92, 96, 100, 105
2	5, 12, 15, 17, 37, 56, 67, 69, 70, 71, 77, 80, 92, 96, 100, 105
3	5, 12, 15, 17, 37, 49, 67, 69, 70, 71, 77, 80, 92, 96, 100, 105
4	5, 12, 15, 17, 32, 37, 49, 56, 59, 71, 77, 80, 85, 92, 96, 100
5	5, 12, 15, 17, 32, 37, 49, 56, 59, 69, 71, 77, 80, 92, 96, 100

noise. Therefore, the impacts of noisy PMU measurements are analyzed. Following the study in [46], and [47], white Gaussian noise is added to the simulated PMU data. Different signal-to-noise ratios (SNRs) of 50dB, 60dB, 70dB, and 80dB are considered. The classification accuracies under these noise levels, presented in Fig. 8, show that noise only degrades the performance of *GAN*1. Larger noise will lead to lower classification accuracy in *GAN*1. However, the classification accuracy of *GAN*2 and *GAN*3 is not affected by noise.

Interestingly, *GAN*1 achieves the highest classification accuracy when a noise of 80dB is added to the measurements. This accuracy is even higher than that obtained using clean PMU data. This observation can be explained by the fact that adding a small amount of noise to the input data leads to a better and more robust GAN performance, as more features of the transient data are learned for classification [48].

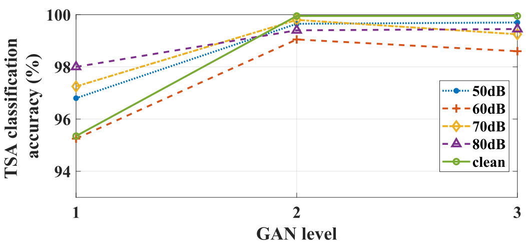


FIGURE 8. **TSA classification accuracy of each sub-GAN under different noise conditions. The testing data are used for evaluation.**

In practice, PMUs are not deployed at all locations within the power grids. The performance of the developed TSA is evaluated under varying PMU placement scenarios, listed in Table 6. The classification accuracies listed in Table 7 show that the location of PMUs significantly impacts the performance of *GAN*1. Due to the benefits offered by the predicted data, the impact of placement is reduced in *GAN*2 and *GAN*3. The accuracy decreases slightly in *GAN*3 due to the small accumulation of errors in the lower GANs.

The impact of the number of PMUs is demonstrated in Table 8. Increasing the number of PMUs improves the TSA accuracy for *GAN*1. The classification accuracy of *GAN*2 and *GAN*3 are around 99% for more than 12 PMUs. However, with less than 10 PMU, the classification accuracy of *GAN*2 and *GAN*3 drops, indicating that the predicted data from the lower GANs are not close to the real data. The prediction error propagates to higher GANs and increases the cumulative error, degrading the classification accuracy.

TABLE 7. **The TSA classification accuracy of each GAN with different PMU locations. A total of 16 PMUs are used.**

PMU Location Scenario	TSA Classification Accuracy (%)		
	*GAN*1	*GAN*2	*GAN*3
1	94.55	98.55	97.90
2	97.15	99.40	98.40
3	96.65	99.65	99.15
4	97.90	98.55	97.55
5	95.40	98.25	97.20

TABLE 8. **TSA classification accuracy of each sub-GAN with different numbers of PMUs.**

Number of PMUs	*GAN*1	*GAN*2	*GAN*3
6	86.90%	84.65%	88.20%
8	91.20%	91.15%	77.95%
10	94.20%	94.55%	95.20%
12	95.75%	99.60%	99.65%
14	96.05%	99.93%	99.92%
16	96.65%	99.65%	99.15%
18	96.25%	99.40%	99.91%
20	95.35%	99.95%	99.95%

While more PMU data improves the classification accuracy, after a threshold, it remains fixed in higher-level GANs. Nevertheless, to ensure a robust TSA, it is recommended that redundant PMUs be deployed throughout the grid.

The prediction accuracy of the sequence data relies on each sub-GAN. The error in lower-level GANs propagates to higher-level GANs and results in a large accumulated error. To improve the prediction, we suggest: (1) freezing the lower-level GANs when training the higher-level GANs to avoid overfitting; (2) adding an additional encoder and decoder to the generator to better predict the sequence data. The training dataset is made balanced, i.e., the number of stable and unstable transients is close. In real world, the ratio between unstable and stable events is less than 5%, since unstable transients are rare. An imbalanced training dataset can adversely affect classification accuracy. To address this, two strategies are suggested: data resampling and redesigning the classifier algorithm. Data resampling, such as oversampling and undersampling, increases the ratio between the two types of events [49]. The classifier algorithm can also be adjusted to be more sensitive to unstable events by assigning a higher cost to an incorrect prediction of unstable events [50].

D. DISCUSSIONS

Alongside GAN model enhancements, incorporating additional transient monitoring and prediction tools into grid operations is critical. Although the developed GAN-based TSA demonstrates a high prediction accuracy, the risk of false predictions, which could potentially compromise grid reliability, remains a concern. Therefore, it is suggested to incorporate the GAN-based TSA approach with other

TABLE 9. **TSA classification accuracy against post-fault Topology Change and 8-cycle fault clearing time.**

	TSA classification Accuracy	
Method	8-cycle clearing time	Topology Change after fault clearing
Decision Tree	80%	83.33%
SVM	76.67%	70%
LSTM	90%	53.3%
GRU	53.3%	50%
HGAN	96.67%	86.67%

real-time monitoring technologies, such as Wide Area Monitoring Systems (WAMS) [51] and advanced real-time simulation tools [52], to provide a robust and redundant transient event assessment.

Additionally, this study utilizes a 5-cycle fault clearing time, with post-clearing breaker reclosure enabled. Results demonstrate the GAN-based TSA method's effectiveness with a single PMU data sample under varying system conditions, topologies, and numbers of PMUs with respective noise ratios. However, this study also acknowledges the variability of fault clearing times and the absence of auto-reclosing on some lines. Therefore, further evaluations on the IEEE 118-bus system are performed for different clearing times and scenarios without auto-reclosing. Specifically, 30 transient events (15 stable and 15 unstable) with an 8-cycle clearing time are generated. These events simulate three-phase faults at the middle of a transmission line with one line out of service. To evaluate the impact of post-fault topology changes, 30 additional transient events (15 stable, 15 unstable) without auto-reclosing are simulated. Moreover, four baseline methods, i.e., Decision tree, SVM, LSTM, and GRU, are evaluated using the same test data. The TSA classification accuracy for each method is given in Table 9. It can be observed that the developed HGAN achieves the highest classification accuracy of 96.67% under 8-cycle clearing time and 86.67% under topology change after fault clearing. However, in these evaluations, the HGAN exhibit reduced prediction accuracy under 8-cycle fault clearing and topology change scenarios, compared to the initial 99.95% classification accuracy. Similarly, the other four baseline methods also show a decline in classification performance under these scenarios. This classification accuracy decrease is due to the training process focusing primarily on scenarios with auto-closing enabled and a standard 5-cycle fault clearing time. Variations in post-fault conditions, such as longer fault clearing times or topology alterations, are found to influence stability results, thus impacting the prediction accuracy of these TSA methods. Future improvements will focus on training the TSA model with a more complex contingencies, including diverse fault clearing times and changes in topology after fault clearance.

V. CONCLUSION

A new GAN-based TSA is developed for the post-fault transient assessment of a power system. Unlike the conventional GAN model, the generator is redesigned such that the modified GAN model predicts the sequence data from near-real-time measurements. Taking advantage of the specific hierarchical structure of multiple GANs, the HGAN-based model maintains the temporal features of the multivariate PMU time-series data. Therefore, with only one sample of PMU measurements, the developed TSA achieves high accuracy. The ability to utilize only one sample of data to generate accurate predictions of system transients is a significant achievement compared to conventional methods. Case studies show that TSA accuracy can reach 99.95% using predicted data. Compared to decision trees, SVM, GRU, and LSTM, and the stacked-GRU model, the developed TSA achieves higher accuracy in a shorter time. It is also observed that the HGAN-based TSA is robust to measurement noise, location, and number of PMUs.

In the future, the robustness of the classifier to imbalanced training datasets can be improved. An encoder and decoder can also be added to the generator to further improve the predicted transient data and help predict the stability margins. Additionally, more complex grid scenarios, such as different clearing time and post-fault topology, will be studied to refine the developed TSA approach.

REFERENCES

[1] A.-A. Fouad and V. Vittal, *Power System Transient Stability Analysis Using the Transient Energy Function Method*. London, U.K.: Pearson Education, 1991.

[2] D. R. Gurusinghe and A. D. Rajapakse, "Post-disturbance transient stability status prediction using synchrophasor measurements," *IEEE Trans. Power Syst.*, vol. 31, no. 5, pp. 3656–3664, Sep. 2016.

[3] T. L. Vu and K. Turitsyn, "Lyapunov functions family approach to transient stability assessment," *IEEE Trans. Power Syst.*, vol. 31, no. 2, pp. 1269–1277, Mar. 2016.

[4] M. Pai and P. W. Sauer, "Stability analysis of power systems by Lyapunov's direct method," *IEEE Control Syst. Mag.*, vol. 9, no. 1, pp. 23–27, Jan. 1989.

[5] E. Farantatos, R. Huang, G. J. Cokkinides, and A. P. Meliopoulos, "A predictive generator out-of-step protection and transient stability monitoring scheme enabled by a distributed dynamic state estimator," *IEEE Trans. Power Del.*, vol. 31, no. 4, pp. 1826–1835, Aug. 2016.

[6] Y. Xue, T. Van Custem, and M. Ribbens-Pavella, "Extended equal area criterion justifications, generalizations, applications," *IEEE Trans. Power Syst.*, vol. 4, no. 1, pp. 44–52, Feb. 1989.

[7] M. Z. Jahromi and S. M. Kouhsari, "A novel recursive approach for real-time transient stability assessment based on corrected kinetic energy," *Appl. Soft Comput.*, vol. 48, pp. 660–671, Nov. 2016.

[8] M. Pai, *Energy Function Analysis for Power System Stability*. Berlin, Germany: Springer, 2012.

[9] P. Ju, H. Li, C. Gan, Y. Liu, Y. Yu, and Y. Liu, "Analytical assessment for transient stability under stochastic continuous disturbances," *IEEE Trans. Power Syst.*, vol. 33, no. 2, pp. 2004–2014, Mar. 2018.

[10] J. Yan, C.-C. Liu, and U. Vaidya, "PMU-based monitoring of rotor angle dynamics," *IEEE Trans. Power Syst.*, vol. 26, no. 4, pp. 2125–2133, Nov. 2011.

[11] M. Li, A. Pal, A. G. Phadke, and J. S. Thorp, "Transient stability prediction based on apparent impedance trajectory recorded by PMUs," *Int. J. Electr. Power Energy Syst.*, vol. 54, pp. 498–504, Jan. 2014.

[12] R. Diao, V. Vittal, and N. Logic, "Design of a real-time security assessment tool for situational awareness enhancement in modern power systems," *IEEE Trans. Power Syst.*, vol. 25, no. 2, pp. 957–965, May 2010.

[13] T. Amraee and S. Ranjbar, "Transient instability prediction using decision tree technique," *IEEE Trans. Power Syst.*, vol. 28, no. 3, pp. 3028–3037, Aug. 2013.

[14] B. Wang, B. Fang, Y. Wang, H. Liu, and Y. Liu, "Power system transient stability assessment based on big data and the core vector machine," *IEEE Trans. Smart Grid*, vol. 7, no. 5, pp. 2561–2570, Sep. 2016.

[15] J. Geeganage, U. D. Annakkage, T. Weekes, and B. A. 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.

[16] Y. Xu, Z. Y. Dong, J. H. Zhao, P. Zhang, and K. Po Wong, ''A reliable intelligent system for real-time dynamic security assessment of power systems,'' *IEEE Trans. Power Syst.*, vol. 27, no. 3, pp. 1253–1263, Aug. 2012.

[17] Y. Li and Z. Yang, ''Application of EOS-ELM with binary Jaya-based feature selection to real-time transient stability assessment using PMU data,'' *IEEE Access*, vol. 5, pp. 23092–23101, 2017.

[18] S. K. Azman, Y. J. Isbeih, M. S. E. Moursi, and K. Elbassioni, ''A unified online deep learning prediction model for small signal and transient stability,'' *IEEE Trans. Power Syst.*, vol. 35, no. 6, pp. 4585–4598, Nov. 2020.

[19] L. Zhu, D. J. Hill, and C. Lu, ''Hierarchical deep learning machine for power system online transient stability prediction,'' *IEEE Trans. Power Syst.*, vol. 35, no. 3, pp. 2399–2411, May 2020.

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

[21] Z. Li, H. Liu, J. Zhao, T. Bi, and Q. Yang, ''Fast power system event identification using enhanced LSTM network with renewable energy integration,'' *IEEE Trans. Power Syst.*, vol. 36, no. 5, pp. 4492–4502, Sep. 2021.

[22] F. Pan et al., ''Stacked-GRU based power system transient stability assessment method,'' *Algorithms*, vol. 11, no. 8, p. 121, Aug. 2018.

[23] Q. Zhu et al., ''A deep end-to-end model for transient stability assessment with PMU data,'' *IEEE Access*, vol. 6, pp. 65474–65487, 2018.

[24] I. J. Goodfellow et al., ''Generative adversarial networks,'' 2014, *arXiv:1406.2661*.

[25] J. Bao, D. Chen, F. Wen, H. Li, and G. Hua, ''CVAE-GAN: Fine-grained image generation through asymmetric training,'' in *Proc. IEEE Int. Conf. Comput. Vis.*, 2017, pp. 2745–2754.

[26] O. Mogren, ''C-RNN-GAN: Continuous recurrent neural networks with adversarial training,'' 2016, *arXiv:1611.09904*.

[27] C. Ren and Y. Xu, ''A fully data-driven method based on generative adversarial networks for power system dynamic security assessment with missing data,'' *IEEE Trans. Power Syst.*, vol. 34, no. 6, pp. 5044–5052, Nov. 2019.

[28] X. Zheng, B. Wang, and L. Xie, ''Synthetic dynamic PMU data generation: A generative adversarial network approach,'' in *Proc. Int. Conf. Smart Grid Synchronized Meas. Analytics (SGSMA)*, May 2019, pp. 1–6.

[29] X. Zheng, B. Wang, D. Kalathil, and L. Xie, ''Generative adversarial networks-based synthetic PMU data creation for improved event classification,'' *IEEE Open Access J. Power Energy*, vol. 8, pp. 68–76, 2021.

[30] R. Ma and S. Eftekharnejad, ''Data generation for rare transient events: A generative adversarial network approach,'' in *Proc. IEEE Ind. Appl. Soc. Annu. Meeting (IAS)*, Oct. 2021, pp. 1–6.

[31] V. Vittal, J. D. McCalley, P. M. Anderson, and A. Fouad, *Power System Control and Stability*. Hoboken, NJ, USA: Wiley, 2019.

[32] *TSAT Transient Security Assessment Tool*, PowerTech Labs Inc, Surrey, BC, Canada, 2007.

[33] T. Guo and J. V. Milanovic, ''Probabilistic framework for assessing the accuracy of data mining tool for online prediction of transient stability,'' *IEEE Trans. Power Syst.*, vol. 29, no. 1, pp. 377–385, Jan. 2014.

[34] G. Gong, N. K. Mahato, H. He, H. Wang, Y. Jin, and Y. Han, ''Transient stability assessment of electric power system based on voltage phasor and CNN-LSTM,'' in *Proc. IEEE/IAS Ind. Commercial Power Syst. Asia (ICPS Asia)*, Jul. 2020, pp. 443–448.

[35] A. D. Rajapakse, F. Gomez, K. Nanayakkara, P. A. Crossley, and V. V. Terzija, ''Rotor angle instability prediction using post-disturbance voltage trajectories,'' *IEEE Trans. Power Syst.*, vol. 25, no. 2, pp. 947–956, May 2010.

[36] C. W. Taylor et al., ''WACS-wide-area stability and voltage control system: R&D and online demonstration,'' *Proc. IEEE*, vol. 93, no. 5, pp. 892–906, May 2005.

[37] J. Yoon, D. Jarrett, and M. van der Schaar, ''Time-series generative adversarial networks,'' in *Proc. 33rd Int. Conf. Neural Inf. Process. Syst. (NIPS)*, Dec. 2019, pp. 5508–5518.

[38] A. Radford, L. Metz, and S. Chintala, ''Unsupervised representation learning with deep convolutional generative adversarial networks,'' 2015, *arXiv:1511.06434*.

[39] M. Frid-Adar, I. Diamant, E. Klang, M. Amitai, J. Goldberger, and H. Greenspan, ''GAN-based synthetic medical image augmentation for increased CNN performance in liver lesion classification,'' *Neurocomputing*, vol. 321, pp. 321–331, Dec. 2018.

[40] L. R. Medsker and L. Jain, ''Recurrent neural networks,'' *Design Appl.*, vol. 5, pp. 64–67, Dec. 2001.

[41] A. Sorjamaa and A. Lendasse, ''Time series prediction using DirRec strategy,'' in *Proc. ESANN*, vol. 6. Princeton, NJ, USA: Citeseer, 2006, pp. 143–148.

[42] C. Zhang and Y. Ma, *Ensemble Machine Learning: Methods and Applications*. Springer, 2012.

[43] S. Basumallik, S. Eftekharnejad, N. Davis, and Brian. K. Johnson, ''Impact of false data injection attacks on PMU-based state estimation,'' in *Proc. North Amer. Power Symp. (NAPS)*, Sep. 2017, pp. 1–6.

[44] R. Ma, S. Basumallik, and S. Eftekharnejad, ''A PMU-based data-driven approach for classifying power system events considering cyberattacks,'' *IEEE Syst. J.*, vol. 14, no. 3, pp. 3558–3569, Sep. 2020.

[45] Y. Wu, M. Kezunovic, and T. Kostic, ''Cost minimization in power system measurement placement,'' in *Proc. Int. Conf. Power Syst. Technol.*, Oct. 2006, pp. 1–6.

[46] A. Al-Digs, S. V. Dhople, and Y. C. Chen, ''Measurement-based sparsity-promoting optimal control of line flows,'' *IEEE Trans. Power Syst.*, vol. 33, no. 5, pp. 5628–5638, Sep. 2018.

[47] Y. Liu, N. Zhang, Y. Wang, J. Yang, and C. Kang, ''Data-driven power flow linearization: A regression approach,'' *IEEE Trans. Smart Grid*, vol. 10, no. 3, pp. 2569–2580, May 2019.

[48] H. Noh, T. You, J. Mun, and B. Han, ''Regularizing deep neural networks by noise: Its interpretation and optimization,'' 2017, *arXiv:1710.05179*.

[49] G. Haixiang, L. Yijing, J. Shang, G. Mingyun, H. Yuanyue, and G. Bing, ''Learning from class-imbalanced data: Review of methods and applications,'' *Exp. Syst. Appl.*, vol. 73, pp. 220–239, May 2017.

[50] B. Krawczyk, M. Woźniak, and G. Schaefer, ''Cost-sensitive decision tree ensembles for effective imbalanced classification,'' *Appl. Soft Comput.*, vol. 14, pp. 554–562, Jan. 2014.

[51] S. Meliopoulos, G. Cokkinides, R. Huang, E. Farantatos, S. Choi, and Y. Lee, ''Wide area dynamic monitoring and stability controls,'' in *Proc. IREP Symp. Bulk Power Syst. Dyn. Control VIII (IREP)*, Aug. 2010, pp. 1–8.

[52] M. D. Omar Faruque et al., ''Real-time simulation technologies for power systems design, testing, and analysis,'' *IEEE Power Energy Technol. Syst. J.*, vol. 2, no. 2, pp. 63–73, Jun. 2015.

RUI MA received the B.S. degree in electrical engineering from Central South University, Changsha, China, in 2014, the M.S. degree from The Ohio State University, Columbus, OH, USA, in 2015, and the Ph.D. degree from Syracuse University, Syracuse, NY, USA, in 2022. His research interests include power system stability analysis, protection, PMUs in smart grids, application of AI, and cyber-security analysis.

SARA EFTEKHARNEJAD (Senior Member, IEEE) received the B.Sc. degree in electrical engineering from the University of Tehran in 2006, the M.Sc. degree from West Virginia University, WV, USA, in 2008, and the Ph.D. degree in electrical engineering from Arizona State University, AZ, USA, in 2012. She is an Associate Professor with the Department of Electrical Engineering and Computer Science, Syracuse University, Syracuse, NY, USA. Her research focuses on integrating renewable energy resources, uncertainty quantification in power grids, and power system stability with high penetration of renewables.

CHEN ZHONG received the B.S. degree in information engineering from Beijing Institute of Technology, China, in 2014, the M.S. degree in electrical engineering from the Stevens Institute of Technology, in 2016, and the Ph.D. degree from Syracuse University, Syracuse, NY, USA, in 2022. Her research interests include the areas of wireless communication and networking, signal processing, and machine learning.