Day-ahead renewable scenario forecasts based on generative adversarial networks

Congmei Jiang, Yongfang Mao, Yi Chai, Mingbiao Yu.

Abstract

With the increasing penetration of renewable resources, such as wind and solar, the operation and planning of power systems, especially in large-scale integration, are faced with great risks due to the inherent stochasticity of natural resources. Although this uncertainty is anticipated, their timing, magnitude and duration cannot be predicted accurately. In addition, the renewable power outputs are correlated in space and time and bring further challenges in characterizing their behaviors. To address these issues, this paper provides a data-driven method to forecast renewable scenarios considering its spatiotemporal correlations based on generative adversarial networks (GANs), which has the ability to generated realistic samples from an unknown distribution making them one of the hottest areas in artificial intelligence research. We first utilize GANs to learn the intrinsic patterns and model the dynamic processes of renewable energy sources. Then by solving an optimization problem, we are able to generate large number of day-ahead forecasting scenarios. For validation, we use power generation data from NREL wind and solar integration data sets. The experimental results of this present research accord with the expectations.

Index terms: artificial intelligence, unsupervised learning, generative models, renewable energy, scenario generation.

I. Introduction

To protect the environment and reduce consumption of conventional energy resources, renewable energy will become progressively more important as time goes on. However, steaming from the reasonable worries about the negative impacts of the intermittent and unpredictable renewable power on power system reliability and security, additional reserves and facilities are required to accommodate the power imbalance and network transmissions especially to the large-scale integration [1-2]. One widely used approach to capture the uncertainties in renewable resources is by using a set of time-series scenarios, which play an important role in stochastic optimization problems such as unit commitment, trading strategy, energy storage sizing, etc. [3-5]. Therefore, accurate modeling for renewable output is key to increase economic benefits and enforce reliability criteria for decision-making under uncertainty faced by power system participants and operators.

In the literature, scholars have conducted extensive research for generating scenarios. In [6-7], Gaussian copula is used to generate statistical scenarios that accounts for both the interdependence structure of prediction errors and the predictive distributions from wind power probabilistic forecasting. In [8], A moment matching technique is presented to generate scenarios for multivariate random variables with

specified moments and correlations, Cholesky decomposition and various transformations are applied to satisfy the specified correlations. In [9], auto regressive moving average (ARMA) and Monte Carlo simulation are used to generate wind power scenarios. In [10], a scenario generation methodology based on artificial neural networks (ANNs) is proposed to create more representative scenarios for electric load, photovoltaic (PV) and wind production. These approaches have been applied to a single site or an aggregate data set and some of them may be extended to apply to a multisite data set for capturing spatial correlation.

For generating spatial correlated scenarios, time series models [11,12] are illustrated to produce a set of plausible scenarios characterizing the uncertainty associated with wind speed at different geographic sites. In [13], in order to reduce the forecasting error, an ensemble of scenarios is generated from different scenario generation algorithms, including support vector machine, multilayer perception, regularized linear regression and random forests. The spatial correlation is characterized by the information regarding the geographical location of the wind farms. In order to characterize interdependence structure of multivariate stochastic processes, Gaussian copula method is widely used [14] [15] [16]. In [17], The Pair-copula theory, which allows to establish complex dependent structures without restriction of Copula families or parameters, is proposed to wind power uncertainty modeling for the spatial relevance for multiple wind farms.

Despite the substantial advances, scenario generation remains a challenging problem. The variation in different seasons, the dynamic and time-varying nature of weather conditions, the nonlinear and bounded power conversion processes, and the complex spatial and temporal interactions make most of these methods difficult to apply and hard to scale in practice. In addition, one of the biggest problems of scenario generation is the difficulty of modeling and learning the underlying stochastic processes that drives renewable power generation. The generated scenarios cannot represent the intrinsic patterns and realistic time-series of real historical observations of renewable energy resources.

As a branch of unsupervised learning techniques in machine learning, generative models are a powerful approach to learn any kind of data distribution and have achieved tremendous success in just few years. All types of generative models aim at learning the true data distribution of the training set so as to generate new data points with some variations. The most common deep neural network based generative models are generative adversarial networks (GANs) [18], variational autoencoders (VAEs) [19], and generative moment matching networks (GMMNs) [20]. In [21], a model-free, data-driven and scalable approach is proposed for generating renewable scenarios by deep generative models. This is the first work applying GANs for generating realistic scenarios to capture the spatiotemporal correlations of renewable energy resources. The scenarios can also be generated based on specific characteristics (e.g., high wind day, intense ramp events, or large forecasts errors) by using label information in the training process. In [22], Bayesian information is incorporated into the GANs to produce scenarios with different variance and mean value that capture different salient modes in the data. Even if wind and solar data are intentionally mixed, the generators can simultaneously distinguish and generate the respective wind and PV scenarios. In [23], we proposed an improved GAN to generate realistic scenarios for wind power using an alternative technique to impose the Lipschitz constraint in the training procedure. The proposed method can better capture the data distribution of real historical observations and achieve faster convergence to reduce the training time for the generative models. Besides, the method is also less prone

to overfitting for cases where there is an insufficient amount of training data. In [24], VAE is used to generate scenarios for wind and PV power. The generated scenarios are used to the coordinated optimization for hydro-wind-solar integrated systems. In [25], a scenarios generation method using conditional VAE is proposed for renewable scenario generation. The generated scenarios can correctly capture the temporal, spatial, and fluctuant characteristics of historical observations.

These methods of using deep generative models for scenario generation can learn to capture the uncertainty in renewable power production with a full diversity of behaviors, and can be trained through the use of differentiable networks without the need for any additional tuning. Note that all these methods for scenario generation are mainly related to generating scenarios which reflects the intrinsic dynamics/patterns of renewable energy sources and cannot generate a group of future scenarios which reflect both forecast information and dynamic patterns. With high penetration of wind and solar power integration, the uncertainty forecasting for renewable power generation through forecast series can be extremely valuable to a number of power system operation and management procedures, including but not limited to, e.g. economic generation scheduling and dispatch, optimal operation of combined wind-storage systems, system steady-state security assessment, electricity market trading, and so forth. In [26], an approach based on unsupervised deep learning for scenario forecasts is proposed to generate a group of future realizations. The generated scenarios can capture the reliability and sharpness features and reflect both forecast information and dynamic patterns of volatile power production. However, this method cannot generate forecast times-series for spatially correlated multiple sites. Due to the similarity of meteorological conditions, outputs of different locations have a natural correlation. The spatial dependence is imperative for joint uncertainty modeling, especially for power flow optimizations and transmission risk assessments.

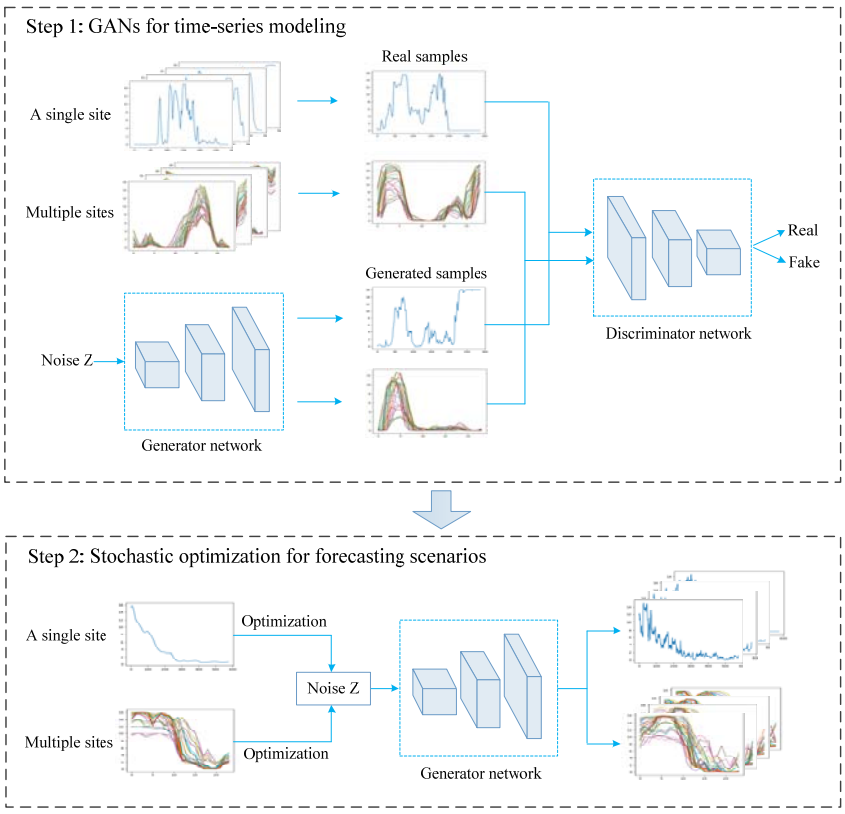


Fig 1: Illustration of the framework for different forecasting tasks based on GANs.

To overcome above issues, based on the previous research in [21], [23] and [26], we proposed a novelty method to generate trajectories for uncertainty forecasting of renewable power generation. Fig. 1 shows the framework for the proposed method. Our proposed method for scenarios forecasts contains two steps. Based on deep learning, the GANs used in our proposed method are unsupervised learners who can directly learn and generate time-series which hold the same properties as the training data. A generator network for fitting the data distribution, and a discriminator network for judging whether the input is "true". In the training process, the generator network tries to "cheat" discriminator network by receiving a random noise to imitate the real sample in the training set, and the discriminator network tries to distinguish the real data and the output of the generator network as much as possible, thus forming the game process of the two networks. Ideally, the outcome of the game would be a generative model that could be “falsely true”. Once training is completed from step 1, we are able to optimize over the noise vectors to find the future scenarios from generator outputs. The following optimization step 2 would help us find a group of scenarios conditioned on forecast information. Specifically, the contributions of the present paper can be briefly summarized as follows:

1) Based on any provided point forecasts along with historical observations, our method is able to generate a group of day-ahead forecasting scenarios representing spatially correlations of stochastic generation dynamics. To our knowledge, this is the first work that applies deep generative models for forecasting spatiotemporal scenarios.

2) Our method can forecast time-series trajectories for wind and solar energy without any changes to the model structure and algorithms for different scenarios forecasting tasks. It also has high flexibility on number of renewable generation sites and scenarios and is not limited to forecasting day-ahead scenarios.

3) Compared with existing method for forecasting scenarios, the network capacity of the discriminator can be better utilized in the training and the data distribution of real historical observations can be better captured by the generator network. The adversarial training can avoid the problems of exploding or vanishing gradients and can also achieve faster convergence to reduce the training time for the generative models. Besides, the proposed deep neural networks are less prone to overfitting for cases where there is an insufficient amount of training data. More details can be found in the previous work in [21], [23] and [26].

The rest of this paper is organized as follows. Section II gives the description of the GANs and the related improvement theories, as well as the model training effects for renewable energy sources. In section III, the setup is detailed for the optimization problem with pre-trained GANs. Section IV provides the model structure and training algorithms for forecasting renewable scenarios. In section V, the numerical and graphical results are illustrated to test the proposed technique through a comprehensive analysis comprising forecasting scenarios for a single site and spatial correlated multiple sites. Finally, in Section VI, some relevant conclusions are duly drawn.

II. Generative Adversarial Networks

A GAN offers a new methodology to draw realistic samples from an unknown distribution with the promise of utilizing large volumes of unlabeled training data for unsupervised learning making them one of the hottest research areas in machine learning/artificial intelligence. Since the introduction by Goodfellow in 2014, GAN has received great attention and have been used in various applications [27-31]. However, the original GAN has problems such as training instability, lack of diversity in generating

samples, and the loss of generator and discriminator cannot indicate the training process [29], [32-34]. Wasserstein GAN (WGAN) [32] is considered to be an effective alternative for traditional GAN training. WGAN can improve the stability of learning, get rid of problems like mode collapse, and provide meaningful learning curves useful for debugging and hyperparameter searches. In terms of the application in our research work, a variant of GANs called WGAN is used to learn the data distribution of historical renewable power generation. In this section, we first introduce the WGAN and related theories. With the improved training, we then can use an advanced GAN variant called Wasserstein GAN with a consistency term (WGANCT) [34] to implement the task for modeling the time-series data for renewable energy sources.

1) Wasserstein GAN

A GAN is a two-player zero-sum game between two interconnected neural networks (i.e. the generator G and the discriminator D) under the adversarial learning idea. The generator's goal is to find a function that transforms a well-defined noise distribution to a sample following the same distribution as the historic observations. The discriminator's goal is to distinguish whether the input data comes from the generator or real samples. When the adversarial networks are trained to an equilibrium, the discriminator can no longer distinguish between generated and historical data, which means the generator can produce realistic samples as if they are coming from the true distribution.

Suppose the distribution of the historical data x is represented by the probability density function P_r, and a noise vector z is sampled from a given distribution P_z, such as uniform distribution or Gaussian distribution. The generator is trained to fool the discriminator to output plausible samples. With the objective defined, we need to formulate a loss function L_G to update the weights of G's neural network. During the training, a batch of samples drawn with distribution P_z are fed into G. Then G outputs newly generated data whose distribution obeys P_G. A small L_G can be achieved by maximizing D(G(z)), which indicates the generated samples from distribution P_G are looking like real samples for the discriminator. Following this guideline, the loss function L_G can be expressed as

$$L_G = -\mathrm{E}_{z\sim \mathrm{P}_z}[D(G(z))] \tag{1}$$

The discriminator takes input samples either coming from generator or coming from real historical data. It is alternately trained with the generator. During the training, the discriminator' goal is to distinguish between P_x and P_G, in other words, to maximize the value between $\mathrm{E}[D(\cdot)]$ and $\mathrm{E}[D(G(\cdot))]$. To update the weights of D's neural network, the loss function L_D can be similarly defined by Eq. (2). A small L_D can be attained via maximizing $D(x)$ and minimizing $D(G(z))$, which reflects the discriminator is good at telling the difference between input samples.

$$L_D = -\mathrm{E}_{x\sim \mathrm{P}_r}[D(x)] + \mathrm{E}_{z\sim \mathrm{P}_z}[D(G(z))]. \tag{2}$$

As for the adversarial training of the two interconnected neural networks, the discriminator outputs a continuous value to measure the input samples. For a given D, maximized output $G(z)$ means to minimize $-\mathrm{E}[D(G(\cdot))]$, resulting in the loss function in (1). On the other hand, for a given G, the discriminator wants to minimize $\mathrm{E}[D(G(\cdot))]$ (generated samples), and at the same time maximize $\mathrm{E}[D(\cdot)]$ (real samples). This gives the expression in (2). With the two loss functions L_D and L_G defined, we then can formulate the two-player game with a value function $V(G,D)$:

$$\min_G \max_D V(G,D) = \mathrm{E}_{x\sim \mathrm{P}_r}[D(x)] - \mathrm{E}_{z\sim \mathrm{P}_z}[D(G(z))]. \qquad (3)$$

More formally, the minimax objective (3) of the game can be interpreted as the dual of the so-called Wasserstein distance, also known as Earth-Mover (EM) distance [35], [36]. In terms of mode training, this distance has nicer properties when optimized than other metrics (e.g., Jensen-Shannon divergence, Kullback-Leibler divergence). The equation of Wasserstein distance is shown as follows:

$$W(P_r, P_G) = \inf_{\psi \in \prod(P_r, P_G)} \mathrm{E}_{(x,y)\sim\psi}[\|x-y\|], \qquad (4)$$

where $\prod(P_r, P_G)$ denotes the set of all joint distributions $\psi(x,y)$ whose marginals are respectively P_r and P_G. Intuitively, $\psi(x,y)$ indicates how much "mass" must be transported from x to y in order to transform the distributions P_r into the distribution P_G. The EM distance then is the "cost" of the optimal transport plan. However, the objective function in such formula is impractical to be achieved by the neural networks. Thanks to the Kantorovich-Rubinstein duality [35], it turns out that

$$W(P_r, P_G) = \sup_{\theta_D} \mathrm{E}_{x\sim \mathrm{P}_r}[D(x)] - \mathrm{E}_{z\sim \mathrm{P}_z}[D(G(z))], \qquad (5)$$

where θ_D is the parameter of discriminator network.

2) Improved training procedures

WGAN uses Wasserstein distance to measure the distance between the generated data distribution and the real data distribution, theoretically solving the problem of unstable training. But it sometimes can still generate low-quality samples or fail to converge in some settings. The change of metric requires the weights of the discriminator to lie within a compact space to enforce the Lipschitz constraint. Since the capacity of the network is limited by the weight constraint, it is really a huge waste of discriminator's own powerful fitting ability. If the clipping parameter c is not carefully tuned, the optimization process will also can result in either vanishing or exploding gradients.

Therefore, an improved strategy is proposed for imposing the Lipschitz constraint [33]. Inspired by the optimal discriminator that has unit gradient norm almost everywhere under P_G and P_r, the gradient penalty is given by

$$GP|_{\hat{x}} = \mathrm{E}_{\hat{x}\sim \mathrm{P}_{\hat{x}}}[(\| \nabla_{\hat{x}} D(\hat{x}) \|_2 - 1)^2], \quad (6)$$

where $\hat{x} = tx + (1-t)G(z)$ for $t \sim U[0,1]$.

Given that enforcing the unit gradient norm constraint everywhere is intractable, this alternative way is an effective way to use for mode training. With the gradient term GP explicitly defined, the new objective is

$$L_{GP} = \mathrm{E}_{z\sim \mathrm{P}_z}[D(G(z))] - \mathrm{E}_{x\sim \mathrm{P}_r}[D(x)] + \lambda GP|_{\hat{x}}. \quad (7)$$

The gradient penalty term GP performs better than the standard weight clipping for Lipschtiz constraint. The modified loss function stabilizes the GAN training over a wide range of architectures (e.g., DCGAN architecture and 101-layer ResNet) with almost no hyper-parameter tuning and can generate higher quality samples on different datasets (e.g., CIFAR-10 and LSUN bedrooms).

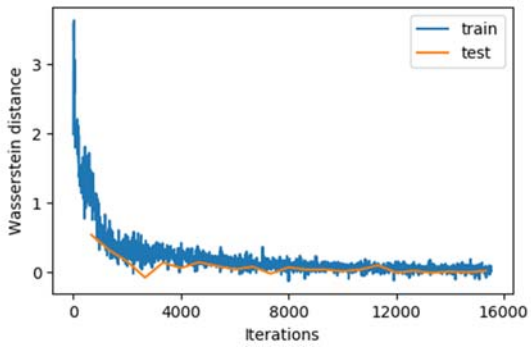
Since the gradient term can only be punished at sampled data points in the training process, a large part of the data points will not be sampled at all. In addition, the output of the generator is significantly different from the actual data point at the start of the training. The 1-Lipschtiz constraint is not enforced until the data distributions P_G and P_r are close enough to each other. To overcome these issues, an additional consistency term (CT) [34] is proposed to improve the training. Instead of focusing on particular data points sampled on specific data points, a region around the real data manifold is considered. In particular, two perturbed data points x' and x'' near observed real data point x are used to check the continuity condition. The two virtual points are found by applying the stochastic dropout to the hidden layers of the discriminator. The performance can be slightly improved by further controlling the second-to-last layer $D_{\_}(\cdot)$ of the discriminator. The final consistency regularization takes the following form,

$$CT|_{x',x''} = \mathrm{E}_{x\sim \mathrm{P}_r}[\max(0, d(D(x'), D(x'')) + 0.1 \cdot d(D_{\_}(x'), D_{\_}(x'')) - M')] \quad (8)$$

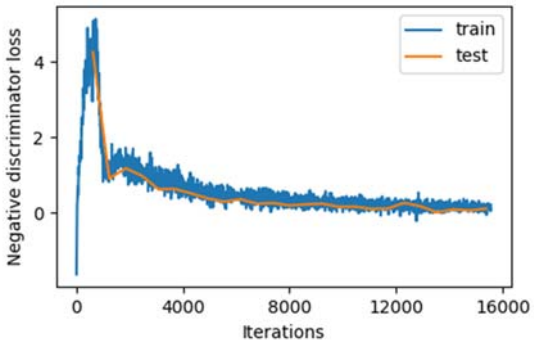
where M' is a bounded constant and d denote the ℓ_2 metric on an input space.

The gradient penalty term GP, (6), enforces the continuity over the points sampled between the real and generated points, while the consistency term CT, (8), can complement the former by focusing on the region around the real data manifold instead. Therefore, these two terms can be used together to improve the training of GANs. Putting them together, the new objective function L_D can be expressed as

$$L_{CT} = \mathrm{E}_{z\sim \mathrm{P}_z}[D(G(z))] - \mathrm{E}_{x\sim \mathrm{P}_r}[D(x)] + \lambda_1 GP|_{\hat{x}} + \lambda_2 CT|_{x',x''}. \quad (9)$$



(a)



(b)

Fig. 2: Training evolutions for GANs on (a) a solar dataset and (b) a wind dataset, respectively.

With the redefined value function L_{CT}, we then can train generator to capture the data distribution of historical observations. An important benefit of WGANs is that the training result is continuously approximated by training the discriminator to optimality, which provides a useful convergence metric for research on adversarial training. To show that our method preserves this property, we train WGAN on NREL renewable integration dataset and plot the evolution of the loss function of the discriminator in Fig. 2. By training the generator and the discriminator to an equilibrium, we see that the two loss curves of wind and solar power gradually converge to a minimum and remain stable. In order to further check whether the discriminator overfits to avoid lead to provide an inaccurate estimate of training point at which all bets are correlated with sample quality. We further explore the loss curve's behaviors on test set and plot the negative discriminator losses in Fig. 2. We can see that the discriminator losses of the test sets (orange curves) consistently decrease with the almost same trend with that (blue curves) of the training sets, which demonstrates that the generative models are well trained. Once the training completed, we get an optimal generator that can capture the data distribution in the true realizations. In the next part, we will introduce how to use the pre-trained GANs for scenario forecasts.

III. Forecasting Scenarios using GANs

Since the uncertainty forecasting of a single site is a special case of multiple sites, for simplicity, we give the problem formulation for multiple sites. For a typical multiple renewable power generation sites,

assume at timestep t, we have some forecasting method to obtain the point forecasts $\hat{\mathrm{p}}_{i,j}$ for each power generation site i and each look-ahead time j, i = 1,…, K, j=1,…, T. This forecast can be denoted by

$$\hat{\mathrm{p}}_{pred} = \begin{bmatrix} \mathrm{p}_{1,1} & \mathrm{p}_{1,2} & \cdots & \mathrm{p}_{1,\mathrm{T}} \\ \mathrm{p}_{2,1} & \mathrm{p}_{2,2} & \cdots & \mathrm{p}_{2,\mathrm{T}} \\ \vdots & \vdots & \ddots & \vdots \\ \mathrm{p}_{K,1} & \mathrm{p}_{\mathrm{K},2} & \cdots & \mathrm{p}_{\mathrm{K,T}} \end{bmatrix} \tag{10}$$

where K denotes the number of sites and T is the forecasting horizon. It should be noted that the size of the predicted information should be resized to match the sample size of the training set.

In this paper, we focus on the scenario forecasting problem, so the central point forecast can be provided by any method, e.g., information from numerical weather prediction (NWP). Assume we have trained a GANs model based on the dataset of observations. Given some input noise z, the pre-trained $G(z)$ generates a possible realization without regarding to the forecast information $\hat{\mathrm{p}}_{pred}$. Based on the generator and the point forecast, we are interested in generating a group of N scenarios $S = \{\mathrm{s}_1, ..., \mathrm{s}_N\}$, which represent the uncertainty of renewable generation and accurately reflect the temporal and spatial dynamics of future generation.

we use the point forecast $\hat{\mathrm{p}}_{pred}$ by defining a prediction interval that the generated scenarios should lie in [26,37]. We describe this interval with an upper bound $U_\theta(\hat{\mathrm{p}}_{pred})$ and a lower bound $L_\theta(\hat{\mathrm{p}}_{pred})$, controlled by a parameter θ (can be interpreted as the prediction confidence or prediction interval):

$$L_\theta(\hat{\mathrm{p}}_{pred}) = \frac{1}{\theta}\hat{\mathrm{p}}_{pred}, \qquad U_\theta(\hat{\mathrm{p}}_{pred}) = \theta\,\hat{\mathrm{p}}_{pred} \tag{11}$$

Since the forecasting scenarios should reflect the forecast information around the point forecast $\hat{\mathrm{p}}_{pred}$, we can first obtain a starting point for z by solving the following problem:

$$\begin{aligned} \min_{z} \quad & \left\| \mathbb{P}_{pred}(G(z)) - \mathrm{p}_{init} \right\|_2 \\ s.t. \quad & z \in Z \\ & L_\theta(\hat{\mathrm{p}}_{pred}) \le \mathrm{p}_{init} \le U_\theta(\hat{\mathrm{p}}_{pred}). \end{aligned} \tag{12}$$

where p_{init} is sampled uniformly at random from an initial fluctuation interval $[L_\theta(\hat{\mathrm{p}}_{pred}),\ U_\theta(\hat{\mathrm{p}}_{pred})]$.

Note that our goal is to forecast scenarios that not only can represent the uncertainty of future time, but also can generate realistic time-series that can capture the intrinsic patterns of renewable energy sources at different prediction horizons. According to the loss defined in (1), larger discriminator output indicates more realistic samples. To ensure the generated scenarios are realistic with pre-trained generator, we use the following objective function (also known as loss or cost function):

$$\min_{z} \quad -D(G(z)) \tag{13}$$

Meanwhile, we want to constrain generated scenarios within a pre-determined confidence interval θ according to actual needs of risk management. Using all of the objectives above and pre-trained model G, D, the scenario forecasts problem can be formulated as a constrained optimization problem:

$$\begin{aligned} \min_{z} \quad & -D(G(z)) \\ s.t. \quad & z \in Z \\ & L_{\theta}(\hat{\mathrm{p}}_{pred}) \le \mathbb{P}_{pred}(G(z)) \le U_{\theta}(\hat{\mathrm{p}}_{pred}) \end{aligned} \tag{14}$$

In order that we can always obtain a good initial z, we set θ in (12) to be slightly smaller than θ in main objective function (14). Since both of the objective and constraints in (14) are nonconvex, to deal with the inequality constraints, we propose to substitute it into the main objective with two log barriers. Then the optimization problem is reformulated as

$$\begin{aligned} \min_{z} \quad & -\lambda D(G(z)) - \tau(\mathbb{P}_{pred}(G(z)) - L_{\theta}(\hat{\mathrm{p}}_{pred})) - \upsilon(U_{\theta}(\hat{\mathrm{p}}_{pred}) - \mathbb{P}_{pred}(G(z))) \\ s.t. \quad & z \in Z \end{aligned} \tag{15}$$

where λ, τ, υ are the weighting parameter.

Since there are multiple local optima to (12), we can start at different initial points $z_i \in Z$ and find distinct forecasting scenario $P_{pred}(\mathrm{G}^{*}_{z_i})$ by solving (15). As the training loss defined in (1) incurs G to generate diverse modes given different z, we are able to obtain a group of distinct yet realistic scenarios that not only can reflect the point forecast information, but also can represent different uncertainty levels according to the actual needs of risk management.

IV. Network structure and training details

GANs have flexible network structure. The framework for GANs is to formulate the generative modeling problem as an adversarial process that is based on two interconnected deep neural networks. In this section, the network structure and the algorithms are described.

A. Network Structure

The network structure of the GANs is based on our previous work in [23]. The generator network starts with fully connected multilayer perceptron and 3 de-convolutional layers to up-sample the input noise z to generate renewable time-series. The discriminator network has a reverse structure to distinguish data from historical samples and generated samples with a single sigmoid output. Sigmoid is an activation function and is used to limit the output range in the interval [0,1]. ReLU activation and LeakyReLU activation is respectively used in the hidden layers of the generator and the discriminator. Dropout is only applied in the output of each hidden layer of the discriminator. Batch normalization can be used to help stabilize training in both the generator and the discriminator, but it changes the form of the discriminator's problem from mapping a single input to a single output to mapping from an entire batch of inputs to a batch of outputs. Since the improved loss objective in (9) is no longer valid in this setting, we can omit or replace the batch normalization by layer normalization [38] in our model structure.

B. Algorithms

Our proposed method for scenario forecasts contains two steps as shown in Fig. 1. The time-series modeling of renewable energy sources can be trained using Algorithm 1. The adversarial networks learn the data distribution of historical data in a batch updating style with a mini-batch size of 64. We use Adam optimizer to update the parameters of the discriminator network and the generator network. In our experiment, we use λ_1 = 10 from [33] and λ_2 = 2 from [34] for the setting of GANs. Another hyperparameter M' from the consistency term CT can take a value between 0 and 0.2. In all experiments, n_{critic} is set to 5, so that there are 5 numbers of discriminator iterations per generator iteration in the alternative training of the adversarial networks. Once the model is trained to convergence by using this algorithm, the generator is able to generate renewable power profiles that preserve the same data distribution as historical observations.

In Algorithm 2 we summarize our approach for forecasting a group of scenarios for step2 as shown in Fig. 1. This algorithm contains two parts. We should first find a good initial z according to the point forecasts. Then we feed this z to the generator to generate time-series trajectories according to different PI level θ. This parameter can be set according to actual needs. Momentum and RMSprop are the optimization algorithms that have been the most reliable for a long time and are suitable for different deep learning structures. Since Adam combines the advantages of these two popular optimization methods and is robust and well-suited to a wide range of non-convex optimization problems in the field machine learning [39]. Therefore, we use Adam for the gradient-based optimization of stochastic objective functions. With pre-trained G, D, we are able to easily obtain a large number of distinct yet realistic scenarios. All our experiments for forecasting scenarios are programed using Python 3.6 with an open source software library TensorFlow [40].

Algorithm 1 GANs for time-series modeling

Input: the batch size m, weights λ_1, λ_2, the learning rate γ, number of iterations N_{iter}, the number of discriminator iterations per generator iteration n_{critic}.

Initialize: initial parameters θ_d for discriminator and θ_g for generator.

for N_{iter} of training iterations **do**

for n_{critic} of iterations **do**

Update parameter for Discriminator

for i = 1 , …, m **do**

Sample data x~ P_r, latent variable z ~ P_z, a random number $\varepsilon \sim U[0,1]$.

$$\hat{x} \leftarrow \varepsilon x + (1-\varepsilon) G(z)$$

$$L^{(i)} = D(G(z)) - D(x) + \lambda_1 GP|_{\hat{x}} + \lambda_2 CT|_{x',x''}$$

---

end for

$\theta_d \leftarrow Adam(\nabla_{\theta_d} \frac{1}{m}\sum_{i=1}^{m} L^{(i)}, \theta_d, \gamma)$

end for

Update parameter for Generator

Sample a batch of latent variables $\{z^{(i)}\}_{i=1}^{m} \sim \mathrm{P}_z$

$\theta_g \leftarrow Adam(\nabla_{\theta_g} \frac{1}{m}\sum_{i=1}^{m} -D(G(z^{(i)}))L^{(i)}, \theta_g, \gamma)$

end for

Algorithm 2 Proposed GANs for forecasting Scenarios

Input: PI level θ, weighting parameters λ, τ, υ, initial iterations n_{init}, scenario finder iterations n_{scen}, learning rate η, measurements p_{hist} p_{hist}, point forecast $\hat{\mathrm{p}}_{pred}$, scenario number N.

Initialize: Pre-trained GANs model weights θ_g, θ_d.

Generated scenarios $S \leftarrow 0$.

for *iteration* = 0, …, N **do**

Sample p_{init} ~ Unif($L_\theta(\hat{\mathrm{p}}_{pred})$, $U_\theta(\hat{\mathrm{p}}_{pred})$)

Sample z ~ Unif (-1,1)

Find good initial z

for *iteration* = 0, …, n_{init} **do**

Update z using gradient descent:

$g_z \leftarrow \nabla_z L_{sub}$ # L_{sub} is defined by 7

$z \leftarrow z - \eta \cdot Algorithm3(z, g_z)$

$z \leftarrow clip(z, -1, 1)$

end for

Find forecasting scenarios

for *iteration* = 0, …, n_{scen} **do**

Update z using gradient descent:

$g_z \leftarrow \nabla_z L_{main}$ # L_{main} is defined by 6

$z \leftarrow z - \eta \cdot Algorithm3(z, g_z)$

$z \leftarrow clip(z, -1, 1)$

end for

S.insert(G(z))

end for

V. EXPERIMENTS

In this section, we describe our experiments and results on a renewable dataset. We show that the proposed method can forecast scenarios for a single site and spatial correlated multiple sites. We validate the effects of the forecasted trajectories in different ways through a comprehensive analysis. These experimental results indicate that using our method would provide an efficient and flexible fashion for scenario forecasts of renewable energy resources.

A. Data Description

In order to test the performance of our proposed framework for scenario forecasts, we build training and validation dataset using power generation data from NREL Wind and Solar Integration Datasets [41]. NREL develops data and tools for the analysis of grid technologies and strategies, including renewable resource data sets and models of the electric power system. Historical power measurements have a resolution of 5 minutes. We choose 24 wind farms and 32 solar power plants located in the State of Washington to use as the training and validating datasets. For different uncertainty modeling tasks, the input samples are divided into training set and validation set. In general, we can randomly select 80% of the input samples as the training set. we also collect the corresponding 24-hour ahead forecast data, which is later used for forecasting scenarios based on pre-trained GANs. All renewable power measurements and forecasts are normalized to [0, 1].

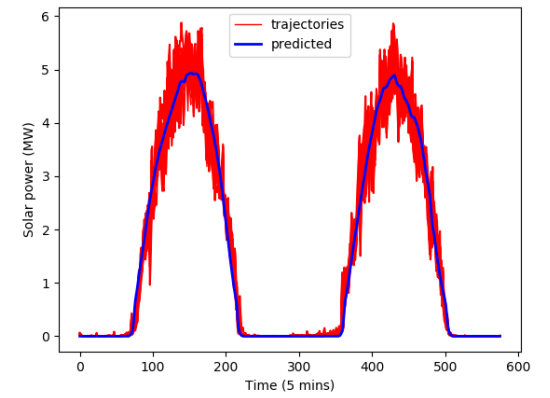
B. Scenario Forecasts

For different scenario forecasting tasks, we can use the same GAN model. The framework for using GANs for forecasting scenarios is illustrated in Fig. 1. The proposed method contains two steps. We can use step 1 to model the uncertainty and capture the data distribution of renewable resources. By solving an optimization, we can generate a large number of forecasted trajectories. In this subsection, we validate the proposed method that can forecasting scenarios for a single site of renewable resources. Historical data in geographical proximity is collected as input samples to represent the stochastic generation dynamics for a single site.

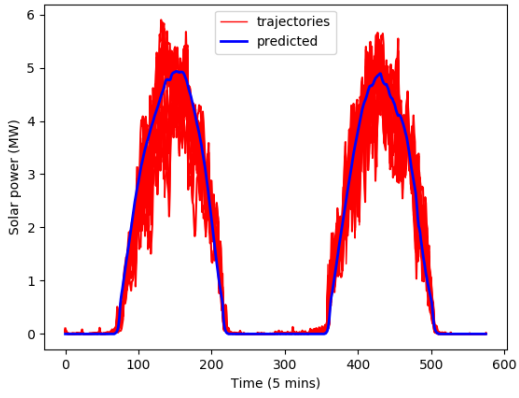
We first show that the proposed method can forecast different levels of uncertainty for solar power. The size of samples from the training set is composed of two-day data with time resolution of 5 minutes. Our generating model is repeatedly inputted with the historical samples until the discriminator loss to converge. We keep the training until about 16 000 iterations to demonstrate the training procedure is stable. The training curves for GANs are shown in Fig. 2(a). At the start, L_D is large because the generator has not yet learned the data distribution of solar power generation. In this case, the generator generates solar scenarios totally different from real observations, and the discriminator can easily distinguish between these scenarios. The generator gradually learns various patterns in historical data.

The generator and the discriminator are continuously updated and alternately trained. After 6000 iterations of training, the loss function shown in Fig. 2(a) already converged to near 0. As the training tends to converge, The generator is able to generate plausible solar power trajectories with a small L_D and the discriminator can hardly distinguish between generated time-series and real ones. Eventually, the output solar power scenarios of the generator can represent the stochastic processes of solar power.

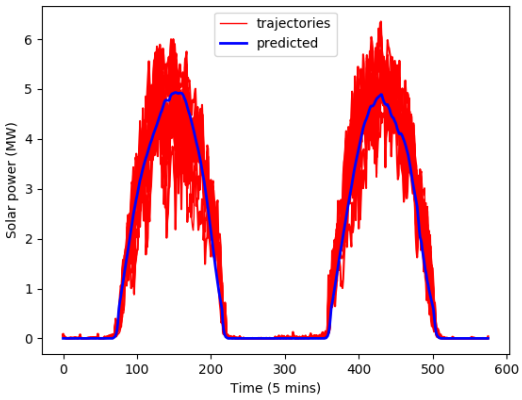
We then use the pre-trained generator to generate time trajectories by Algorithm 2. The forecasted trajectories with varying PIs of 1.5, 2 and 3 are shown in Fig. 3. We can see that the samples generated by proposed methods can correctly capture the hallmark features (e.g., large peak values, daily variations, and ramp events of large fluctuations) of the solar power profiles from the predicted data. By selecting different prediction interval θ, the forecasted trajectories can represent different degrees of uncertainty in solar power generation. The larger θ, the predicted time-series will have larger fluctuations. The prediction interval can be proper selected according to the actual situation. If for power system operation, a larger θ will improve forecast reliability, but reduce operational economics.



(a)



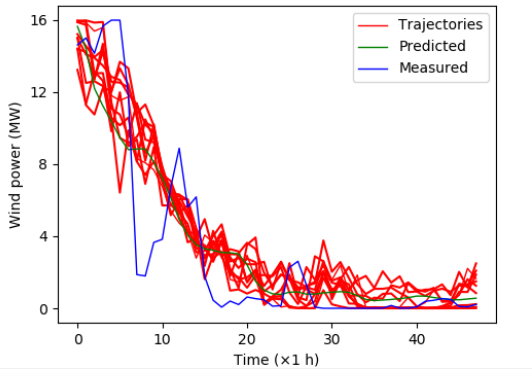
(b)



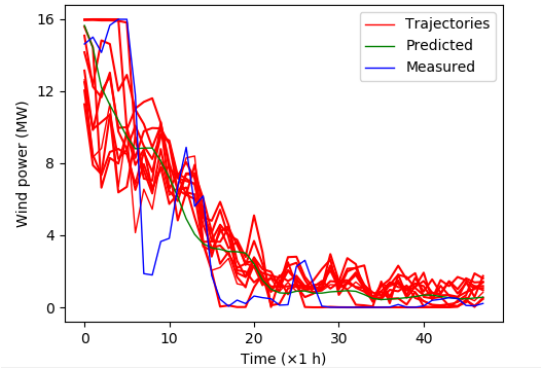
(c)

Fig. 3: Plot (a) (b) (c) correspond to a group of 10 trajectories for solar power with varying PIs of 1.5, 2 and 3 respectively.

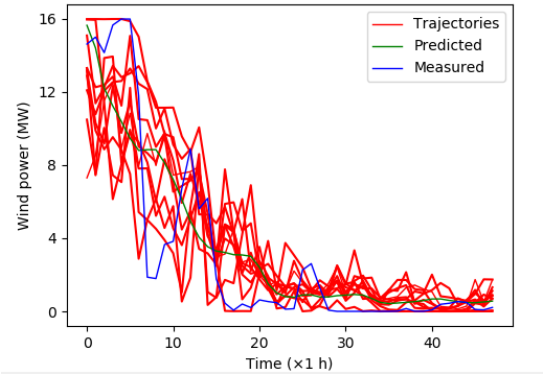
In order to verify the group of generated trajectories are able to represent possible future realizations, the scenarios should be able to cover the actual value of power generation (reliable), while at the same time distance between forecasted scenarios should be small (sharp). Since wind power from NREL renewable dataset has greater volatility than solar power, we use wind power data for presentation. Similarly, we use historic data to train the GANs. We can see the loss function shown in Fig. 2(b) converged to near 0 after 10000 iterations of training. Then we can generate forecasted trajectories with the pre-trained generator by Algorithm 2. In Fig. 4 we specifically select one wind profile whose point forecast is deviating a lot from the actual measurements. We can also observe that the relationship between the prediction interval θ and the prediction uncertainty is similar to that of solar power. By selecting different θ, our proposed method could reflect the trade-off between reliability and sharpness. When the interval level is a = 1.5, generated trajectories are close to point forecasts, yet fail to cover the realizations; while when a = 3, generated trajectories could cover the actual power production values, but are less concentrated. As for the range of the prediction interval θ, it can be adjusted according to the accuracy of information forecasting and the level of risk management. At the same time, we can further adjust the weights τ, υ of the upper and lower boundaries to generate trajectories that is more in line with the actual needs.



(a)



(b)



(c)

Fig. 4: Plot (a) (b) (c) correspond to a group of 10 trajectories for wind power with varying PIs of 1.5, 2 and 3 respectively.

In order to further verify the generated trajectories' temporal statistics, we calculate and compare samples' autocorrelation. The autocorrelation measures the degree of correlation of a time series between two different periods. Since wind power from NREL renewable dataset has greater volatility than solar power, we can use wind power data to present for simplicity. The autocorrelation coefficient $R(h)$ for a wind time-series can be calculated by

$$R(h) = \sum_{i=1}^{n-h} \frac{(s_i - \mu)(s_{i+h} - \mu)}{\sum_{i=1}^{n} (s_i - \mu)^2} \qquad (16)$$

where h is the look-ahead time and s represents generated samples or realizations with mean μ.

The temporal correlation of generated trajectories is shown in Fig. 5. the trajectories' autocorrelation plots cover the predictions, which indicate the generated trajectories are able to represent the temporal dependence of forecast time-series. Autocorrelation represents the temporal correlation at a renewable resource, and capture the correct temporal behavior is of critical importance to power system operations.

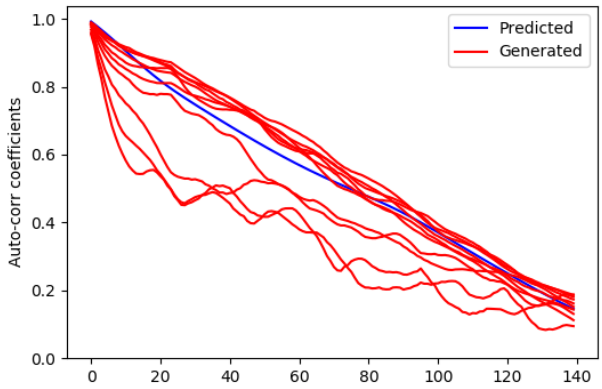


Fig. 5: Autocorrelation plots for both predicted values and generated trajectories.

C. Spatial correlation

For the scenario forecasts of multiple sites, instead of inputting historical data x for a single site, here we input the model with a real data matrix {x} of size K × T, where K denotes the total number of generation sites, while T denotes the total number of timesteps for each scenario. Here we choose K = 24, T = 24 with a resolution of 1 hour. A sample of real scenarios {x(i)} and forecasting scenarios {G(z(i))} for the 24 wind farms are plotted in Fig. 6. By visual inspection we find that their dynamic behaviors are similar to each other. The spatial and temporal correlations in the real data (again, not seen in the training stage) are correctly preserved by our forecasted scenarios. From the spatial correlation coefficient colormaps of these two group of scenarios, we can see that all the patches of these two sets of colormaps have relatively large values. It shows that all power generation sites for this sample have a relatively high correlation.

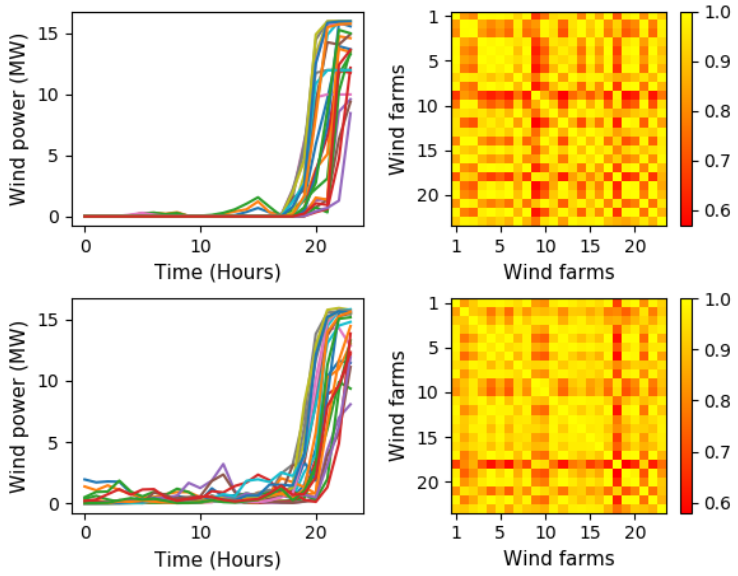


Fig. 6: Wind power scenarios and spatial correlation coefficient colormaps for multiple sites: (top) historical data; (bottom) sample generated by our method.

We also verify that generated time-series have the same statistical properties as the predicted data. We use Algorithm 2 to random generate 50 scenarios. As shown in Figs. 7 and 8, the power generation and

fluctuations at all sites are basically consistent with the predicted data. The level of different generation capacity and the magnitude of different fluctuations can be correctly captured. In probability theory and statistics, the cumulative distribution function (CDF, also cumulative density function) of a real-valued random variable X, or just distribution function of X, evaluated at x, is the probability that X will take a value less than or equal to x. We compare the CDF relevant to the predicted time-series and the generated data in Fig. 9. For the sake of simplicity, we only select some of them for display. It is clear the methodology for different sites has the capability to generate samples with the correct marginal distributions that are basically the same as the predicted time-series.

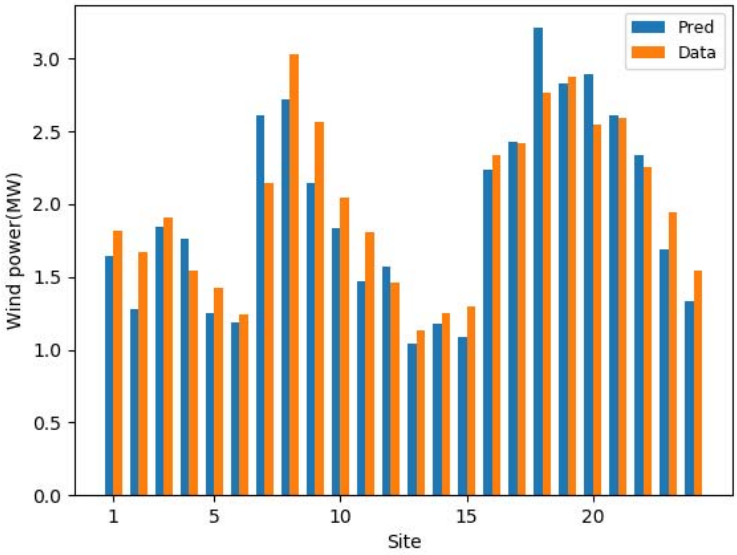


Fig. 7: Wind power mean for the 24 sites.

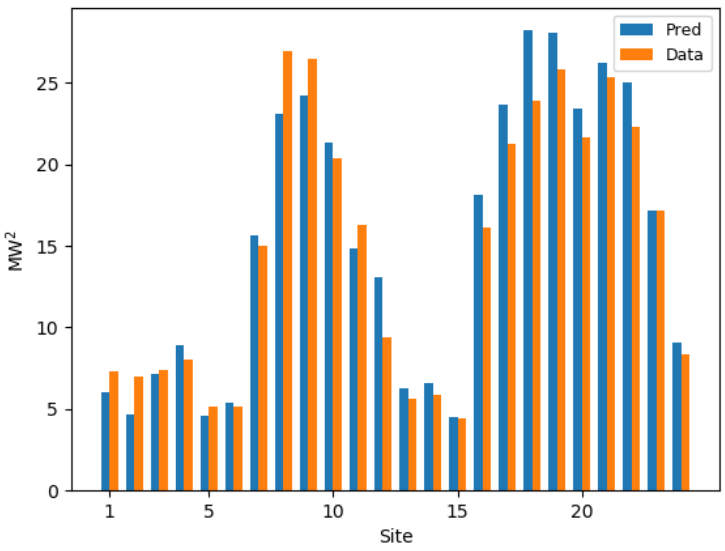


Fig. 8: Wind power variance for the 24 sites.

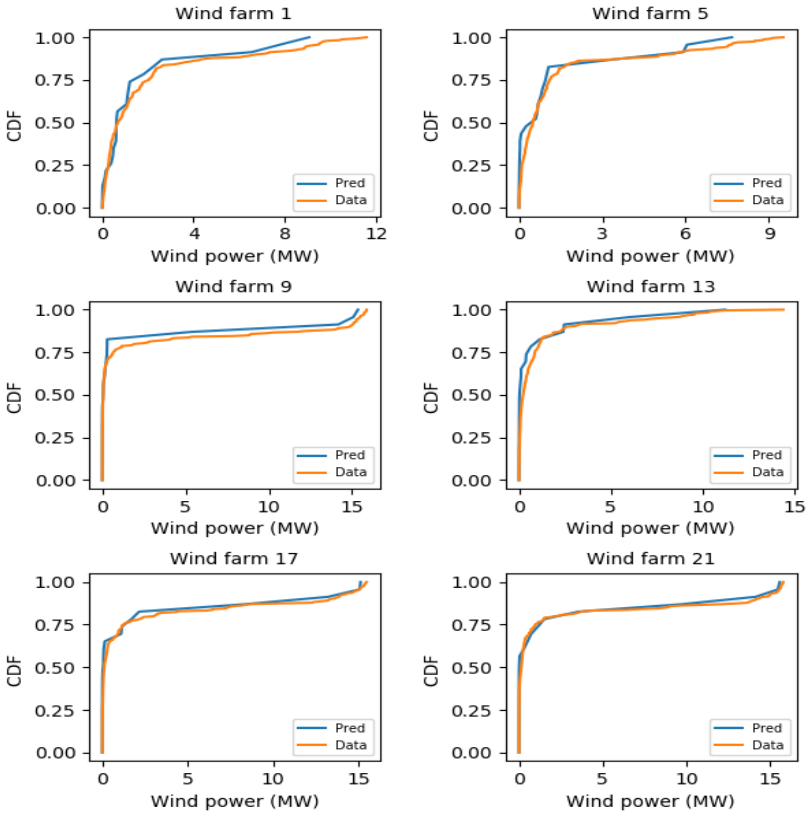


Fig. 9 Cumulative distribution function (CDF) of predicted wind power versus CDF of generated dataset from our trained GANs.

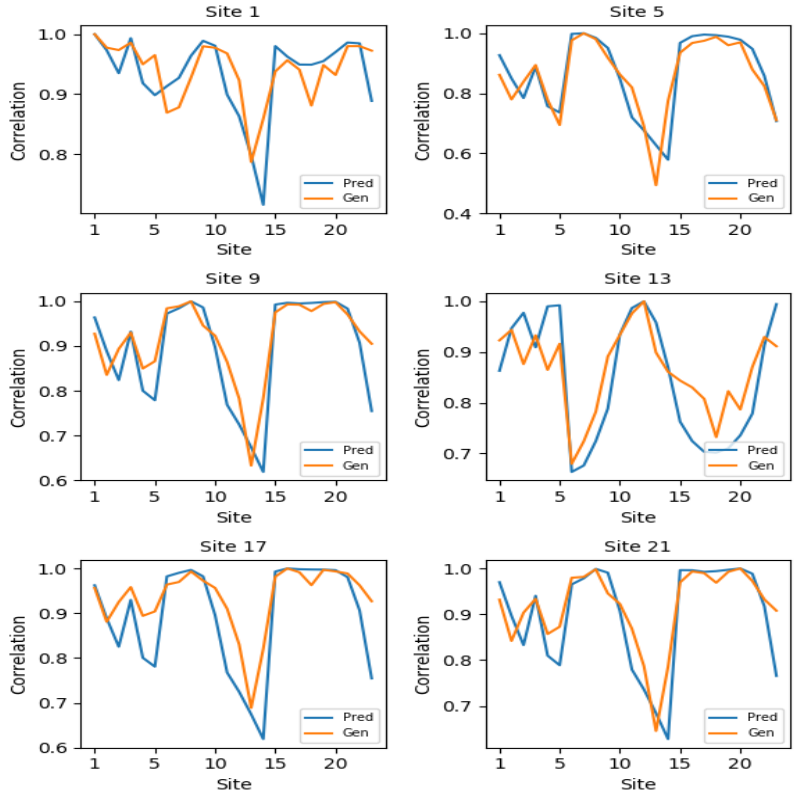


Fig. 10: Correlation between given site and other sites.

In order to further examine the correlations between individual locations, we calculate the correlation matrix of the simulated time series and compares the values with those of predicted time series. Each row of the correlation matrix shows the correlation between that site (e.g. row 1 represents Wind farm 1) and the other sites, so that the diagonal is composed of ones (the site auto-correlation) and the other terms are the cross-correlation between sites. For purposes of illustration, the elements of the correlation matrix for a few sites are shown in Fig. 10, for both the predicted time-series and the model outputs. Each pair of curves basically maintains a relatively consistent trend. The results show that the simulated time series using proposed method agrees with the assigned value, showing that spatial correlations between different sites can be correctly retained.

VI. CONCLUSION

This paper proposes a novelty method to forecast scenarios for renewables power generation processes based on deep generative models. The proposed method can characterize the uncertainty associated with renewable energy sources both for a single site or spatial correlated multiple sites without any changes to the model structure and algorithms.

Our method can not only generate a group of future realizations, but also can generate realistic, high quality time-series the can reflect the intrinsic patterns and data distribution of real observations. The marginal distribution associated with each renewable power stochastic process is retained by the generated times-series. The temporal correlations are characterized by autocorrelations at each renewable stochastic process. The spatial correlations are verified by cross-correlations among different sites. Comprehensive simulations carried out for different case studies show the effectiveness of the proposed methodology. Besides, the method can be easily implemented in problems with high penetration of renewables. With high reliability and high flexibility, the proposed approach can be used to directly generate a large number of time-series and can provide a meaningful tool for uncertainty forecasting in integrated renewable systems.

REFERENCES

[1] Halamay, D.A., et al., “Reserve Requirement Impacts of Large-Scale Integration of Wind, Solar, and Ocean Wave Power Generation,” IEEE Trans. Sustain. Energy, vol. 2, no.3, pp. 321-328, 2011.

[2] Burke, D.J., O’Malley, M.J. “Maximizing Firm Wind Power Connection to Security Constrained Transmission Networks,” IEEE Trans. Power Syst., vol. 25, no.2, pp.749-759, 2010.

[3] Wang J, Shahidehpour M, Li Z. Security-Constrained Unit Commitment With Volatile Wind Power Generation[J]. IEEE Transactions on Power Systems, 2008, 23(3):1319-1327.

[4] X. Ma, Y. Sun, H. Fang and Y. Tian, "Scenario-Based Multiobjective Decision-Making of Optimal Access Point for Wind Power Transmission Corridor in the Load Centers," in IEEE Transactions on Sustainable Energy, vol. 4, no. 1, pp. 229-239, Jan. 2013.

[5] Wang Y, Dvorkin Y, Fernandez-Blanco R, et al. Look-Ahead Bidding Strategy for Energy Storage[J]. IEEE Transactions on Sustainable Energy, 2017, PP(99):1-1.

[6] P. Pinson, H. Madsen, H. A. Nielsen, G. Papaefthymiou, and B. Klockl, From probabilistic forecasts to statistical scenarios of short-term wind power production,” Wind Energy, vol. 12, no. 1, pp. 51–62, 2009.

[7] P. Pinson and R. Girard, “Evaluating the quality of scenarios of short term wind power generation,” Applied Energy, vol. 96, pp. 12–20, 2012.

[8] Høyland K, Kaut M, Wallace S W. A Heuristic for Moment-Matching Scenario Generation[J]. Computational

Optimization & Applications, 2003, 24(2-3):169-185.
[9] P. Meibom et al., “Stochastic optimization model to study the operational impacts of high wind penetrationsin Ireland,” IEEE Trans. Power Syst., vol. 26, no. 3, pp. 1367–1379, Aug. 2011.
[10] S. I. Vagropoulos, E. G. Kardakos, C. K. Simoglou, A. G. Bakirtzis, and J. P. S. Catalao, ``ANN-based scenario generation methodology for stochastic variables of electric power systems," Electr. Power Syst. Res., vol. 134, pp. 9_18, May 2016.
[11] Juan Morales, Roberto Mínguez, and Antonio Conejo, “A methodology to generate statistically dependent wind speed scenarios,” Applied Energy, vol. 87, no. 3, pp. 843-855, 2010.
[12] Le, Duong D., George Gross, and Alberto Berizzi. "Probabilistic modeling of multisite wind farm production for scenario-based applications." IEEE Transactions on Sustainable Energy 6.3 (2015): 748-758.
[13] H. Pandºi¢, Y. Dvorkin, T. Qiu, Y. Wang, and D. S. Kirschen, ``Toward cost-efficient and reliable unit commitment under uncertainty," IEEE Trans. Power Syst., vol. 31, no. 2, pp. 970-982, Mar. 2016.
[14] J. Tastu, P. Pinson, and H. Madsen, “Space-time scenarios of wind power generation produced using a Gaussian copula with parametrized precision matrix,” Tech. Univ. Denmark, Tech. Rep. 2013, 2013.
[15] Daniel Villanueva, Andrés Feijóo, and José Luis Pazos “Simulation of correlated wind speed data for economic dispatch evaluation,” IEEE Transactions on Sustainable Energy, vol. 3, no. 1, pp. 142-149, Jan. 2012.
[16] Tang C, Wang Y, Xu J, et al. Efficient scenario generation of multiple renewable power plants considering spatial and temporal correlations[J]. Applied Energy, 2018, 221: 348-357.
[17] Q. Lu, W. Hu, Y. Min, F. Yuan and Z. Gao, "Wind power uncertainty modeling considering spatial dependence based on Pair-copula theory," 2014 IEEE PES General Meeting | Conference & Exposition, National Harbor, MD, 2014, pp. 1-5.
[18] Goodfellow I J, Pouget-Abadie J, Mirza M, et al. Generative Adversarial Networks[J]. Advances in Neural Information Processing Systems, 2014, 3:2672-2680.
[19] Kingma, D. P. and Welling, M. Stochastic Gradient VB and the Variational Auto-Encoder. In Proceedings of the 2nd International Conference on Learning Representations (ICLR), Banff, AB, Canada, 14–16 April 2014.
[20] Li, Y., Swersky, K., and Zemel, R., Generative moment matching networks, International Conference on Machine Learning, 1718–1727, 2015.
[21] 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. 32653275, May 2018.
[22] Y. Chen, P. Li, and B. Zhang, ``Bayesian renewables scenario generation via deep generative networks," in Proc. 52nd Annu. Conf. Inf. Sci.Syst. (CISS), Princeton, NJ, USA, Mar. 2018, pp. 1-6.
[23] C. Jiang, Y. Mao, Y. Chai, M. Yu and S. Tao, "Scenario Generation for Wind Power Using Improved Generative Adversarial Networks," in IEEE Access, vol. 6, pp. 62193-62203, 2018.
[24] Hongxuan Zhang, Wei Hu, Rui Yu, Maolin Tang, Lijie Ding. Optimized Operation of cascade reservoirs Considering Complementary Characteristics between wind and photovoltaic based on variational auto-encoder, MATEC Web Conf. 246 01077 (2018).
[25] Shouxiang W , Haiwen C , Xiaoping L I , et al. Conditional Variational Automatic Encoder Method for Stochastic Scenario Generation of Wind Power and Photovoltaic System[J]. Power System Technology, 2018.
[26] Y. Chen, X. Wang, and B. Zhang. (2017). ``An unsupervised deep learning approach for scenario forecasts." [Online]. Available:https://arxiv.org/abs/1711.02247
[27] Wang F, Zhang Z, Liu C, et al. Generative adversarial networks and convolutional neural networks based weather classification model for day ahead short-term photovoltaic power forecasting[J]. Energy Conversion and Management, 2019, 181: 443-462.
[28] Liu H, Zhou J, Xu Y, et al. Unsupervised fault diagnosis of rolling bearings using a deep neural network based

on generative adversarial networks[J]. Neurocomputing, 2018, 315: 412-424.
[29] Radford A, Metz L, Chintala S. Unsupervised Representation Learning with Deep Convolutional Generative Adversarial Networks[J]. Computer Science, 2015.
[30] K.Wang, C. Gou, Y. Duan, Y. Lin, X. Zheng, and F.-Y.Wang, ``Generative adversarial networks: Introduction and outlook," IEEE/CAA J. Autom. Sinica, vol. 4, no. 4, pp. 588_598, Sep. 2017.
[31] Creswell A, White T, Dumoulin V, Kai A, Sengupta B, Bharath AA. Generative adversarial networks: an overview. IEEE Signal Process Mag 2018;35:53–65.
[32] Arjovsky M, Chintala S, Bottou L. Wasserstein GAN[J]. arXiv preprint, 2017.
[33] Gulrajani I, Ahmed F, Arjovsky M, et al. Improved Training of Wasserstein GANs[J]. arXiv preprint, 2017.
[34] Wei X, Gong B, Liu Z, et al. Improving the Improved Training of Wasserstein GANs: A Consistency Term and Its Dual Effect[J]. arXiv preprint, 2018.
[35] C´edric Villani. Optimal transport: old and new, volume 338. Springer Science & Business Media, 2008.
[36] Y. Rubner, C. Tomasi, and L. J. Guibas, “The earth mover’s distance as a metric for image retrieval,” International journal of computer vision, vol. 40, no. 2, pp. 99–121, 2000.
[37] C. Wan, Z. Xu, P. Pinson, Z. Y. Dong, and K. P. Wong, “Probabilistic forecasting of wind power generation using extreme learning machine,” IEEE Transactions on Power Systems, vol. 29, no. 3, pp. 1033–1044, 2014.
[38] Ba J L, Kiros J R, Hinton G E. Layer Normalization[J]. arXiv preprint, 2016.
[39] Diederik P. Kingma and Jimmy Lei Ba. Adam: A Method for Stochastic Optimization. International Conference on Learning Representations, pages 1–13, 2015.
[40] M. Abadi, A. Agarwal, P. Barham, et al. Tensorflow: Large-scale machine learning on heterogeneous distributed systems. ar Xiv preprint arXiv:1603.04467, 2016.
[41] C. Draxl, A. Clifton, B.-M. Hodge, and J. McCaa, “The wind integration national dataset (wind) toolkit,” Appl. Energy, vol. 151, pp. 355–366, 2015.