

Received March 31, 2020, accepted April 18, 2020, date of publication April 21, 2020, date of current version May 8, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.2989350

# Modeling Daily Load Profiles of Distribution Network for Scenario Generation Using Flow-Based Generative Network

LEIJIAO GE<sup>1</sup>, (Member, IEEE), WENLONG LIAO<sup>1,2</sup>, (Graduate Student Member, IEEE),  
SHOUXIANG WANG<sup>1</sup>, (Senior Member, IEEE),  
BIRGITTE BAK-JENSEN<sup>3</sup>, (Senior Member, IEEE),  
AND JAYAKRISHNAN RADHAKRISHNA PILLAI<sup>3</sup>, (Senior Member, IEEE)

<sup>1</sup>Key Laboratory of Smart Grid of Ministry of Education, School of Electrical and Information Engineering, Tianjin University, Tianjin 300072, China

<sup>2</sup>Tianjin Xianghe Electric Company, Ltd., Tianjin 300000, China

<sup>3</sup>Department of Energy Technology, Aalborg University, 9220 Aalborg, Denmark

Corresponding author: Wenlong Liao (wenlongliao@cau.edu.cn)

This work was supported in part by the National Natural Science Foundation of China under Grant 51807134 and, in part by the Science and Technology Project of State Grid Corporation of China.

**ABSTRACT** The daily load profiles modeling is of great significance for the economic operation and stability analysis of the distribution network. In this paper, a flow-based generative network is proposed to model daily load profiles of the distribution network. Firstly, the real samples are used to train a series of reversible functions that map the probability distribution of real samples to the prior distribution. Then, the new daily load profiles are generated by taking the random number obeying the Gaussian distribution as the input data of these reversible functions. Compared with existing methods such as explicit density models, the proposed approach does not need to assume the probability distribution of real samples, and can be used to model different loads only by adjusting the structure and parameters. The simulation results show that the proposed approach not only fits the probability distribution of real samples well, but also accurately captures the spatial-temporal correlation of daily load profiles. The daily load profiles with specific characteristics can be obtained by simply classification.

**INDEX TERMS** Daily load profiles, distribution network, generative network.

## I. INTRODUCTION

### A. BACKGROUND

With the increasing penetration of renewable energy and electric vehicles, uncertainties in their output powers pose challenges to all aspects of the power grid, such as planning and operation. Therefore, it is necessary to propose some methods for modeling uncertainty. The scenario generation is a popular method to capture the uncertainty of power load and renewable energy by generating a series of possible daily power load profiles [1]. Taking a large number of daily load profiles as the input of the Newton-Raphson method, the probability distribution of voltage and power loss is obtained, which is of great significance for the economic operation and stability analysis of the distribution network [2]–[4].

The associate editor coordinating the review of this manuscript and approving it for publication was Petros Nicopolitidis<sup>1</sup>.

The core idea of the scenario generation method is to generate new daily load profiles similar to real samples after learning some given real samples. The scenario generation methods can be divided into explicit density models and implicit density models with respect to whether it is necessary to assume the probability distribution of real samples [5]. The explicit density models need to know the probability distribution of the real daily load profiles which are used to fit the parameters in the probability distribution. Then, the new daily load profiles can be generated by using the trained probability distribution [6]–[8]. The explicit density models need to artificially make assumptions about the probability distribution of real samples, which limit the scope of application, since the probability distribution of most real samples is difficult to be accurately described by mathematical formulas. Moreover, it is not universal, because the probability distributions of daily load profiles vary from region to region [9], [10].

The implicit density model does not require explicitly estimate or fit the probability distribution of daily load profiles. The new data can be generated by sampling and training [11]. To this end, the implicit density model has been widely used in various fields, such as pattern recognition, image generation, text translation, and data imputation [12]–[14]. As popular frameworks of implicit density model, the variational automatic encoder (VAE) and generative adversarial network (GAN) are considered to be two distinct paradigms. They have been extensively and independently studied. However, the training process of the GAN is unstable, which makes it difficult to achieve the Nash equilibrium. Besides, it is prone to the disappearance of the gradient [15]. Unlike GAN that learns samples by constructing a generator and a discriminator in the adversarial state, VAE directly calculates the mean square error between generated samples and real samples, which results in poor performance of VAE than GAN [16]. Therefore, VAE and GAN are not suitable for modeling daily load profiles before these shortcomings are overcome. How to accurately model the daily load profiles through the implicit density method is still a challenge. The non-linear independent component estimation (NICE) model is one of the flow-based generative models, which is a well-known generative framework after GAN and VAE. So far, the NICE model has been widely applied to the tasks of image generation and classification, and has shown good performance [17]. There is no report on the use of the NICE model to generate daily load profiles. How to design a NICE model structure suitable for generating daily load profiles needs further study.

## B. LITERATURE REVIEW

Most of the existing methods for modeling daily load profiles belong to explicit density models that first utilize the real samples to fit the assumed probability distribution then sample the daily load profiles from the probability distribution based on the Monte Carlo method. For example, in order to simulate the daily load profiles of electric vehicles (EVs) charging stations, the traffic topology data, road network information, and detailed travel data are used to estimate the probability distribution of EVs in [18]–[20]. The charging curve of each EV is obtained to form the daily load profiles of EVs charging station. Nevertheless, the travel data of EVs and traffic flow information aren't available at most charging stations, which results in these methods being generally used for theoretical analysis and difficult to apply in practice. In [21], a stochastic bottom-up method is proposed to model household load profiles considering the impact of user behavior and financial incentive. While in [22], the domestic load profiles are generated by taking into account the time-related features of the activities such as duration and starting time. In general, these methods mainly have the following two shortcomings: 1) They are not universal. The probability distribution is different for various loads such as EVs, household loads and power station loads. Moreover, the probability distribution is also affected by space, time and electricity consumption

habits. In this case, it needs to establish mathematical models for different loads separately, which is very troublesome. 2) The probability distribution of the real load curve is difficult to be described by a mathematical formula. They can only be approximated by the usual probability distribution.

In order to overcome the shortcomings of the explicit density models, some scholars use the autoregressive integrated moving average model (ARIMA) and autoregressive moving average (ARMA) model to estimate the peak of daily load profiles. However, they are linear models that do not take into account the nonlinear relationship between power loads and other factors. Besides, these models can only be used for the simulation of a single daily load profile, and the correlation among multiple daily load profiles cannot be considered.

Recently, some deep neural networks have been applied to generate daily load profiles. The core idea of VAE is variational inference. The posterior probability is replaced by variational function, and the similarity between the expected output and the decoder output is measured by Kullback-Leibler divergence. The advantage of the VAE is that it can automatically extract the natural features of real samples to model load profiles. In [23], a variational automatic encoder composed of deep convolution networks is proposed to generate load profiles of EVs. In [24], the variational automatic encoder is designed to simulate a large number of times series for wind and photovoltaic powers. The GAN generates data by constructing an antagonistic generator and discriminator. If the structure and parameters of generator and discriminator are set properly and the training process approximates to the so-called Nash equilibrium, the new samples generated by the generator are almost identical with the real samples. In [25] and [26], a deep convolution generative adversarial network is used to capture the spatial-temporal complementary and fluctuant characteristics of wind and photovoltaic power. In [27], a conditional generative adversarial network is trained to output power consumption data considering demand response programs. Compared to the explicit density models and time series methods, the GAN and VAE can capture nonlinear time characteristics without assuming the probability distribution of daily load profiles manually, but they also have some limitations. For example, the GAN has the problem of gradient disappearance or gradient explosion. The VAE can only infer the approximation of the real probability distribution, which results in poor performance for generating data. The advantage of the NICE model is that it can accurately capture the probability distribution of real samples with latent variables due to the reversibility of the transformation function. At present, the NICE model is mainly used for image processing, and its application in other fields is relatively limited. In [28], a generative framework based on the NICE model is proposed for the synthesis and interpolation of images. Similarly, a flow-based generative network is used to model image through sampling and latent variables manipulations in [29]. To accurately predict power load, a conditional flow-based model is proposed to capture the complex temporal dependency of residential load in [30].

In [31], the improved NICE model is proposed to generate high-quality audio synthesis from Mel-spectrograms. Overall, the NICE model shows good performance for generating 2-dimension and 3-dimension data, such as images and videos. However, the NICE model cannot be used directly to generate 1-dimensional daily load profiles. It is necessary to redesign the structure of the NICE model to make it suitable for processing daily load profiles.

### C. PROPOSED METHOD AND KEY CONTRIBUTIONS

In this paper, it is aimed to design a NICE model to generate daily load profiles. The performance of the proposed method will be tested by the actual data set from London. The key contributions of this paper are as follows:

- 1) New technologies: To the best of our knowledge, it is the first time to design the NICE model for generating 1-dimensional daily load profiles. After training, the NICE model can generate daily load profiles that accurately capture the volatility characteristics and spatial-temporal correlation of the real samples.
- 2) Data-driven load profiles generation: The proposed method utilizes historical data to model daily load profiles without artificially assuming the probability distribution of real samples. Besides, it can be applied to model power loads in different regions only by adjusting the structure and parameters.
- 3) Accurate model probability distribution: The NICE model can accurately capture the probability distribution of real samples with latent variables, which is good for capturing nonlinear time characteristics of real samples. The simulation results on actual datasets show that the NICE model has better performance than VAE and GAN.
- 4) Conditional load profiles generation: The conditional daily load profiles with specific characteristics (e.g. heavy load, light load) can be obtained by simply classifying the generated load profiles.

The rest of the paper is organized as follows. Section II designs the NICE model to generate daily load profiles. Section III presents some indicators to evaluate the performance of the algorithms. Section IV discusses the simulations and results. Section V presents conclusions and future work.

## II. METHODOLOGY

### A. THE PRINCIPLE OF NICE

Normally, the generative networks are designed to learn the probability distribution  $X \sim P_{data(x)}$  of real daily load profiles  $x$ . The VAE and GAN don't explicitly learn the probability density function (PDF) of real samples, since it is very hard to calculate latent variables. By contrast, the NICE model solves this hard problem using normalizing flows, which make it possible to efficiently complete many tasks such as density estimation and data generation. As shown in Fig. 1, NICE model maps the probability distribution of real samples to a priori distribution through a series of reversible functions. Each reversible function is an additive coupling

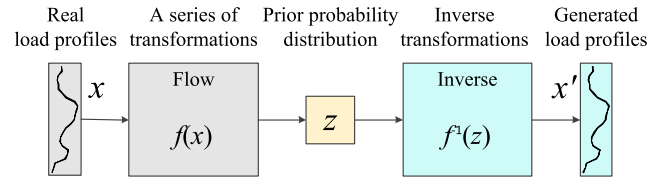


FIGURE 1. A typical framework of NICE.

layer, just like flowing water, so the NICE model is also called a flow-based generative model.

Specifically, a series of reversible functions map the real samples  $x$  to the latent variables  $z$  whose dimensions are the same as that of  $x$ . In order to fit the probability distribution  $Z \sim P_{z(z)}$ , the variable substitution method is used to get the following formula:

$$P_{data}(x) = P_Z(f(x)) \left| \det \frac{\partial f(x)}{\partial x} \right| \quad (1)$$

In formula (1),  $\frac{\partial f(x)}{\partial x}$  is the Jacobian matrix of function  $f$  at  $x$ . The function  $f$  must be reversible and its determinant of the Jacobian matrix must be easily computed. By taking the logarithm of formula (1), we can get the optimization objective:

$$\log(P_{data}(x)) = \log(P_Z(f(x))) + \log \left( \left| \det \frac{\partial f(x)}{\partial x} \right| \right) \quad (2)$$

If the prior distribution can be factorized, then the optimization objective can be rewritten as follows:

$$\log(P_{data}(x)) = \sum_{d=1}^D \log(P_{Z_d}(f_d(x))) + \log \left( \left| \det \frac{\partial f(x)}{\partial x} \right| \right) \quad (3)$$

When the training process is over, the random numbers are obtained by sampling probability distribution  $Z \sim P_{z(z)}$ , and the new daily load profiles are generated by the inverse function  $f^{-1}(z)$ .

In order to make the determinant of the Jacobian matrix easy to be calculated, the NICE model divides  $x$  into two parts which include  $x_1$  and  $x_2$ . It is regarded as a building block to transform from  $(x_1, x_2)$  to  $(y_1, y_2)$ :

$$\begin{cases} y_1 = x_1 \\ y_2 = x_2 + m(x_1) \end{cases} \quad (4)$$

In formula (4),  $y_1$  and  $y_2$  are new variables called additive coupling layers. The  $m$  can be any complex function (e.g. multiple fully connected layers in our model). This building block has the unit Jacobian determinant for any function  $m$ , and its inverse can be expressed as follows:

$$\begin{cases} x_1 = y_1 \\ x_2 = y_2 - m(y_1) \end{cases} \quad (5)$$

Let  $I_1$  be the first  $d$  elements of  $x$  and  $I_2$  be the element of  $x$  from  $d + 1$  to  $D$ .  $D$  is the dimension of  $x$ . The Jacobian

matrix can be rewritten as follows:

$$\frac{\partial y}{\partial x} = \begin{bmatrix} I_d & 0 \\ \frac{\partial y_{I_2}}{\partial x_{I_1}} & \frac{\partial y_{I_2}}{\partial x_{I_2}} \end{bmatrix} = \begin{bmatrix} I_d & 0 \\ \frac{\partial y_{I_2}}{\partial x_{I_1}} & I_{d:D} \end{bmatrix} \quad (6)$$

In formula (6), this Jacobian matrix is a triangular matrix where all the diagonal elements are equal to 1.

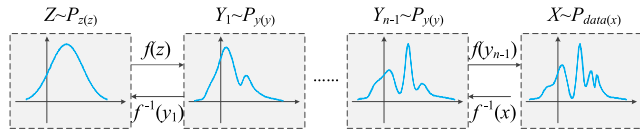


FIGURE 2. The process of the normalizing flow.

It can be seen from formula (4) that although this transformation is reversible, it is relatively simple and difficult to fit very complex non-linear relationships. To this end, the normalizing flow is proposed to estimate the probability distribution of real samples. As shown in Fig.2, the normalizing flow transforms the prior probability distribution  $Z \sim P_{z(z)}$  (e.g. Gaussian distribution) into a complex probability distribution  $X \sim P_{data(x)}$  of daily load profiles by using a series of invertible transformation functions.

From the chain rule and the change of variables theorem, we can get:

$$\frac{\partial z}{\partial x} = \frac{\partial z}{\partial y^{(n-1)}} \times \frac{\partial y^{(n-1)}}{\partial y^{(n-2)}} \cdots \frac{\partial y^{(2)}}{\partial y^{(1)}} \times \frac{\partial y^{(1)}}{\partial x} \quad (7)$$

Further, the determinant of formula (7) can be obtained:

$$\det \left( \frac{\partial z}{\partial x} \right) = \det \left( \frac{\partial z}{\partial y^{(n-1)}} \right) \times \det \left( \frac{\partial y^{(n-1)}}{\partial y^{(n-2)}} \right) \cdots \det \left( \frac{\partial y^{(2)}}{\partial y^{(1)}} \right) \times \det \left( \frac{\partial y^{(1)}}{\partial x} \right) = 1 \quad (8)$$

So far, the problem of calculating determinants has been solved.

The random variable  $z$  has the same dimension as the real sample  $x$ , since the NICE model uses a series of reversible transformations to obtain the probability distribution of daily load profiles. Although the dimension of the real sample  $x$  is  $D$ , it may have a problem of wasting dimensions. For example, the real sample  $x$  has a large number of 0 elements that have no effect. To improve computational efficiency, the NICE model proposes a concept of rescaling. It takes a diagonal scaling matrix  $S$  as the top layer that multiplies the  $i$ -th output data by  $S_{ii}$ . Thus, the formula (3) can be rewritten as follows:

$$\log(P_{data}(x)) = \sum_{i=1}^D \log(P_{z_i}(f_i(x))) + \log(|S_{ii}|) \quad (9)$$

In formula (9), the vector  $S$  can relate these scaling factors to the Eigen spectrum of a PCA by calculating how much variation is present in each of the latent dimensions (the smaller  $S$ , the more important the dimension  $i$  is). More parameters and principles of the NICE model can be found in [32].

### B. THE TRAINING PROCESS OF THE NICE

The training process of the NICE model can be accomplished by using the back propagation algorithm which mainly includes two stages: incentive propagation and weight update [33]. As shown in Fig.3, in the stage of incentive propagation, input information is transmitted to the output layer through layer-by-layer processing of the middle layer. The result of the output layer is used to calculate the loss function for the stage of the weight update. Firstly, the real samples are used to train the parameters  $W_1$  and  $b_1$  in the first middle layer of the NICE model. Then, the output of the first middle layer is used as the input of the next layer, and the parameters  $W_2$  and  $b_2$  in the second middle layer are obtained. By analogy, the network parameters of all layers can be obtained. The result of the output layer is used to calculate the loss function, which is utilized to update the weights of the network.

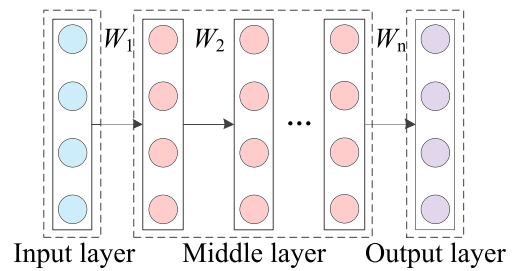


FIGURE 3. The principle of incentive propagation.

In the stage of weight update, its core idea is gradient descent and chain rule. The errors generated by the output layer will gradually propagate to the previous layer, and the weights of each layer will be continuously optimized according to the gradient descent algorithm. Firstly, the loss function is multiplied by the input data to obtain the weights of the gradient. Then, the product of gradient and training factor is inverted to obtain the change of weights that are used to update the old weights.

### C. FORMAT TRANSFORM OF REAL SAMPLES

It is well known that the original intent of designing these deep generative networks is to process images with the same number of rows and columns. However, the daily load profile of the distribution network is a  $1 \times n$  vector, which cannot be used as the input data of these deep generative networks. Therefore, the daily load profiles should be transformed into square matrixes before they are used as input data. The transformation process is shown in Fig.4.

For a single daily load profile generation, an explanation is given by taking 24 sampling points a day as an example. Firstly, the original load profile is transformed into a vector of  $1 \times 25$  scale by adding a 0 element to its end. Then, the vector is transformed into a matrix of  $5 \times 5$  scales by format transformation, which is used as the input data of the NICE model. The data generated by the NICE model is also

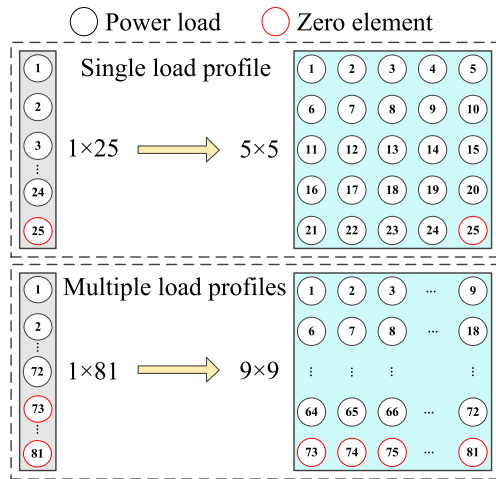


FIGURE 4. An example of format conversion.

a matrix of  $5 \times 5$  scales, and the first 24 values constitute the new daily load profile of the distribution network.

In optimization models of distribution network, the daily load profiles of multiple nodes are often required. For example, the dynamic reactive power optimization model in the IEEE 33 bus system requires 32 daily load profiles of 32 buses. The nodes of the distribution network have similar power consumption in one area, and are prone to simultaneous peaks or valleys. Therefore, the simultaneous generation of multiple daily load profiles considering their spatial correlation is of great significance for distribution network planning. Here, the generation of three daily load profiles is taken as an example for explanation. Firstly, three original load profiles of  $1 \times 24$  scale are merged into a vector of  $1 \times 72$  scale. Then, the vector of  $1 \times 72$  scale is transformed into a vector of  $1 \times 81$  scale by adding 9 zero elements to its end. The vector of  $1 \times 81$  can be transformed into a matrix of  $9 \times 9$  scales by format transformation, which is used as the input data of the NICE model. After training the NICE model, the NICE model can generate a matrix of  $9 \times 9$  scales. We only need to separate 3 daily load profiles from it.

#### D. CONDITIONAL LOAD PROFILES GENERATION

In addition to generating a variety of different daily load profiles, it is sometimes desirable to generate load profiles with specific properties (e.g. heavy load and light load). For example, relay protection devices may need to be tested under heavy and light loads to ensure that signals are properly signaled in the event of a fault. After training the NICE model, the random noises obeying the Gaussian distribution is used as the input data of the NICE model to generate a large number of daily load profiles. The daily load profiles with specific properties can be obtained by simply classifying the generated load profiles.

In this paper, an example is given to illustrate the principle of generating conditional daily load profiles. Firstly, three groups which include light load, medium load, and heavy load are defined. Then, the mean value of each daily load

profile is calculated, and every generated load profile is distributed to one group according to the average value.

#### E. THE PROCESS OF GENERATING DAILY LOAD PROFILES BASED ON NICE

To summarize the previous description, the steps to generate daily load profiles using the NICE model are as follows:

- 1) Data normalization and transformation: Before inputting data into the NICE model, the real samples need to be normalized, and otherwise the loss function may not converge. In this paper, the minimum-maximum normalization method is used to transform the input data to the range of 0 to 1. Then, the time series is transformed into a matrix with the same number of rows and columns.
- 2) Sampling and Mapping: Firstly, a series of reversible functions (e.g. a multi-layer perception in this paper) are used to map the probability distribution of real samples to the prior distribution (e.g. Gaussian distribution in this paper). Then, the daily load profiles are generated by taking the random number obeying the Gaussian distribution as the input data of the NICE model.
- 3) Update parameters: The loss function is calculated using generated data and real data. Then, the weights of the NICE model are adjusted by using the back propagation algorithm.
- 4) Daily load profiles generation: After training, the prior distribution  $Z \sim P_{z(z)}$  is sampled to obtain random numbers, and a large number of daily load profiles are generated by the inverse function (e.g. a multi-layer perception in this paper). The daily load profiles with specific properties can be obtained by simple classification.

#### III. INDICATORS FOR EVALUATING RESULT

In order to evaluate the similarity between the generated daily load profiles and the real samples, 5 indicators are proposed to calculate the probability distribution, spatial-temporal correlation, and volatility of the load profiles.

**Probability density function:** Firstly, several intervals are defined, and the probability that power load falls in some interval is calculated. Then, scatter plots or frequency histograms are drawn to approximate the probability density function of the power loads [34].

**Autocorrelation function:** The autocorrelation function can represent the temporal correlation of daily load profile, which is the correlation of a daily load profile with a delayed copy of itself [35]. The autocorrelation function between time  $t$  and  $t + \tau$  can be expressed as follows:

$$R(\tau) = \frac{E[(X_t - \mu)(X_{t+\tau} - \mu)]}{\sigma^2} \quad (10)$$

In formula (10),  $X_t$  is the daily load profile and  $t$  is any point in time. The daily load profile has a mean  $\mu$  and

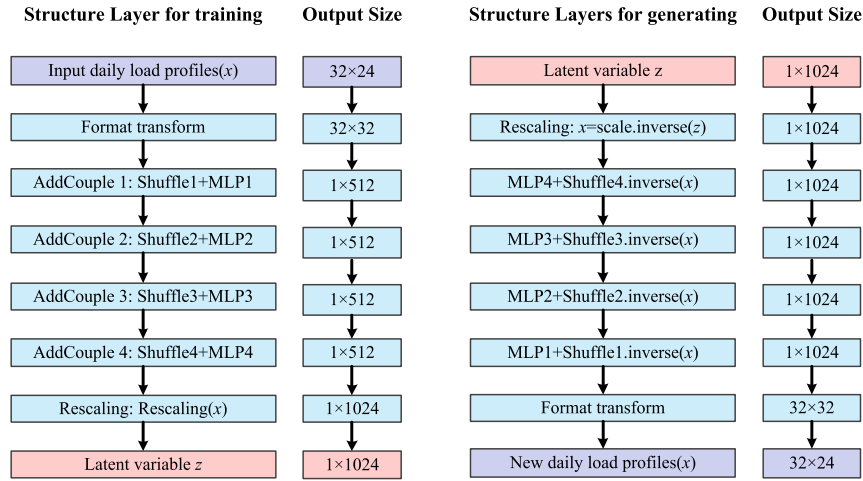


FIGURE 5. Structure and parameters of NICE.

variance  $\sigma^2$ .  $E$  is the expected value operator.  $\tau$  is the lag time.

**Pearson correlation coefficient:** The Pearson correlation coefficient is used to measure the spatial correlation among multiple daily load profiles [36]. Its mathematical formula is as follows:

$$r_{xy} = \frac{\sum_{i=1}^n (x_i - \mu_x)(y_i - \mu_y)}{\sqrt{\sum_{i=1}^n (x_i - \mu_x)^2} \sqrt{\sum_{i=1}^n (y_i - \mu_y)^2}} \quad (11)$$

In formula (11),  $r_{xy}$  is the Pearson coefficient between two daily load profiles.  $\mu_x$  and  $\mu_y$  are the mean of daily load profile  $x$  and  $y$ , respectively.

**Load duration curve:** The load duration curve describes the variation of a certain power load in a downward form that the smallest load is plotted on the right and the greatest load on the left. The area under the load duration curve represents the power consumption per day [15]. The mathematical formula of power load duration is as follows:

$$t_j = \sum_{i=1}^n q_{i,j}, q_{i,j} = \begin{cases} 1, & P_i \geq P_j \\ 0, & P_i < P_j \end{cases} \quad i = 1 \sim n, j = 1 \sim m \quad (12)$$

In formula (12),  $t_j$  is the time when the power loads are greater than  $P_j$ .  $n$  is the length of the daily load profile.  $P_i$  is the  $i$ -th element of the daily load profile.  $m$  is the number of intervals for power load.

**The volatility of daily load profiles:** In addition to mean and standard deviation, the wave rate is defined to evaluate the volatility of the daily load profile. Its mathematical formula is as follows:

$$wave\_rate = quantile(x, 0.9) - quantile(x, 0.1) \quad (13)$$

In formula (13), the wave rate of a daily load profile is the difference between 90-quantiles and 10-quantiles.

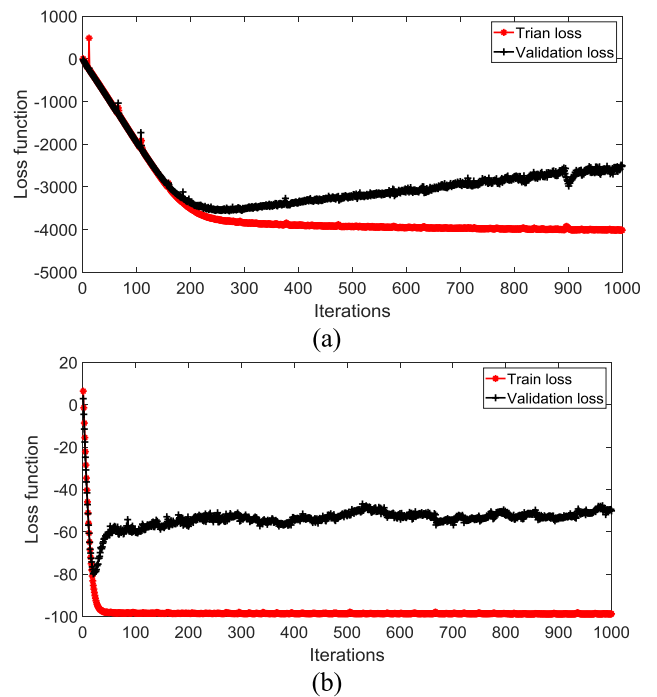


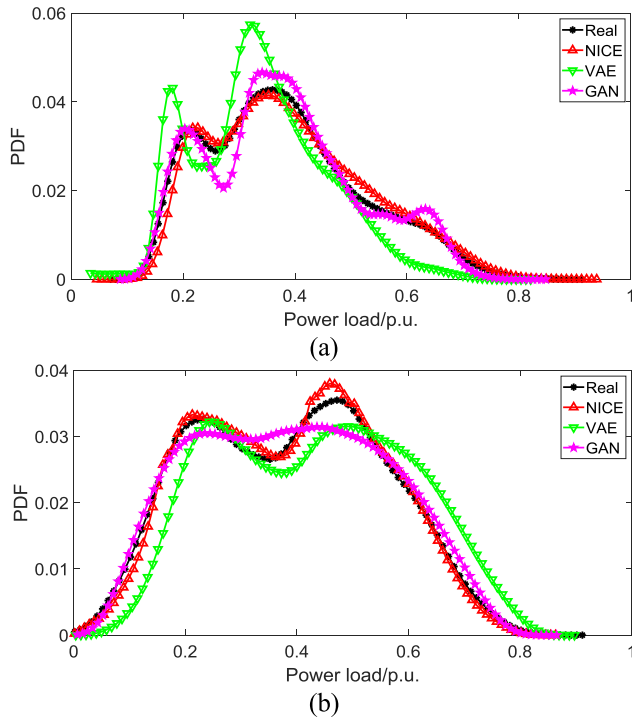
FIGURE 6. Training evolution for NICE: (a) Training evolution of London smart meter data set. and (b) Training evolution of Spanish transmission service operator data set.

## IV. CASE STUDY

### A. DATA DESCRIPTION AND DETAILS OF MODEL

In order to fully verify the effectiveness of the proposed approach, simulations are carried out using the London smart meter data set and Spanish transmission service operator data set from Kaggle competition. The proposed approaches run under Keras that a deep learning library. The configuration parameters of the laptop are: Intel Core i3-3110M, The processor is dual-core 2.4 GHz, 6 GB of memory.

For the London smart meter data set, it counts the hourly electricity consumption of each household in 112 blocks from

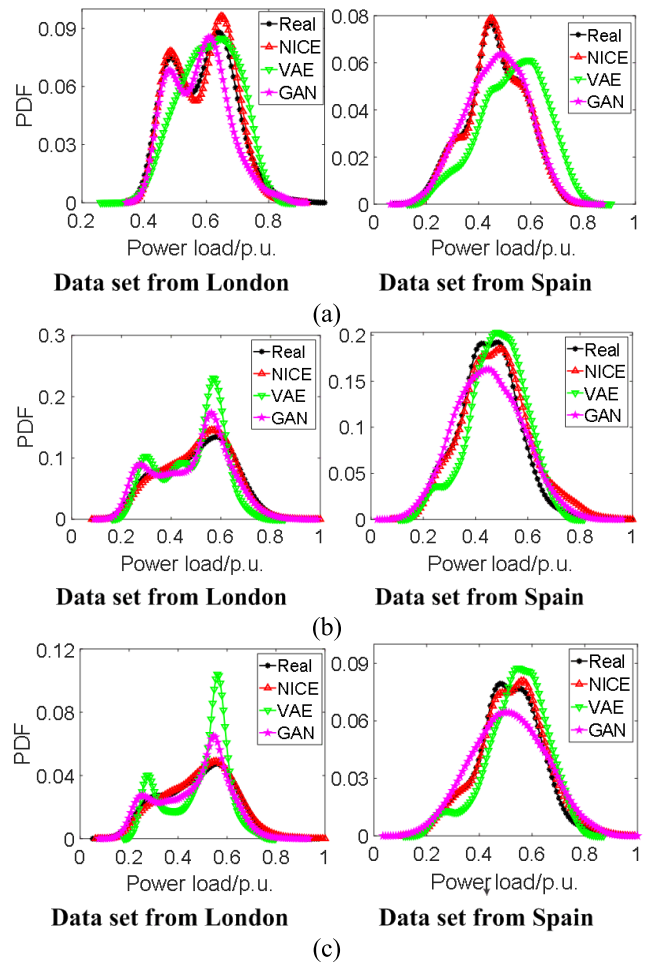


**FIGURE 7.** Probability density functions of daily load profiles: (a) PDF of the London smart meter data set, and (b) PDF of the Spanish transmission service operator data set.

December 2011 to February 2014. Since the statistical time of each block is different, after screening, the data from October 15, 2012 to February 9, 2014 is selected for simulation. The load data from October 15, 2012 to January 9, 2014 are used to train the structure and parameters of the NICE model and the remaining data are used as test sets. The load of three adjacent blocks is combined to analog a node in the distribution network. The first 96 blocks form 32 nodes, and each of which has 483 historical daily load profiles. In terms of format transformation, 32 daily load profiles are merged into a vector of  $1 \times 768$  scale, and then 256 zero elements are added to the end of this vector to form a new vector of  $1 \times 1024$  scale. The new vector is transformed into a matrix of  $32 \times 32$  scales by format transformation, which is used as the input data of the NICE model.

For the Spanish transmission service operator data set, it includes the hourly electricity consumption of one node in the distribution network from January 2015 to December 2018. The load data from January 1, 2015 to March 14, 2018 are used to train the structure and parameters of the NICE model and the remaining data are used as test sets. In terms of format transformation, 1 zero element is added to the end of the daily load profile to form a new vector of  $1 \times 25$  scales. Then, the new vector is transformed into a matrix of  $5 \times 5$  scales by format transformation, which is used as the input data of the NICE model.

After many experiments on the structure and parameters of the NICE model, the optimum parameters are as follows:



**FIGURE 8.** The probability density function of some indicators: (a) PDF of the mean of daily load profile; (b) PDF of the standard deviation of daily load profile and (c) PDF of the wave rate of daily load profile.

The structures of the NICE model for these two datasets are similar. As shown in Fig.5, the NICE model for London smart meter data set is used as an example to illustrate the structure and parameters. The NICE model consists of 4 additive coupling layers where the reversible function is a multi-layer perceptron (MLP). Each MLP consists of four fully connected layers with 1000 neurons. The 4 additive coupling layers map the daily load profiles into the latent variables obeying Gaussian distribution. After training, the random noises obeying Gaussian distribution are taken as the input data of the NICE model. The inverse functions of four additive coupling layers map random noises into new daily load profiles.

In order to test the performance of the NICE model, the VAE and GAN are used as baselines. For the encoder, the number of neurons in the input layer and the middle layer is 1024 and 256 respectively. The output layer consists of two fully connected layers with 2 neurons to obtain the mean and variance. As far as the decoder is concerned, the number of neurons in the input layer, middle layer, and output layer is 2, 256 and 1024 respectively. More parameters and training

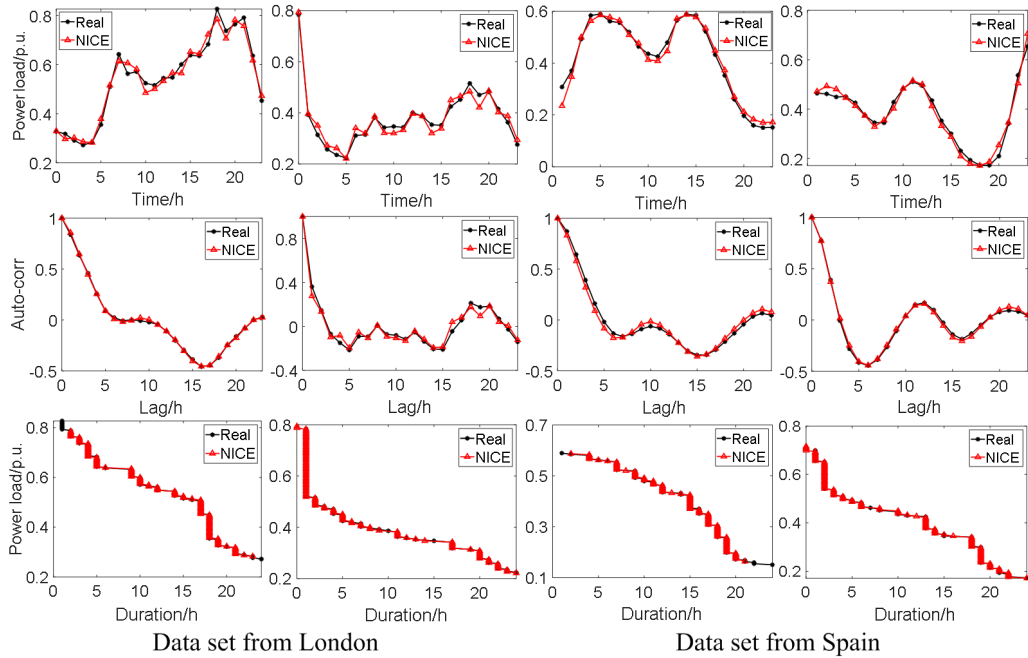


FIGURE 9. The real load profiles, generated load profiles, autocorrelation function, and load duration.

methods of the VAE model can be found in [23]. For GAN, the discriminator consists of 3 fully connected layers, and their numbers of neurons are 512, 256, and 1 respectively. The generator consists of 3 fully connected layers, and their numbers of neurons are 256, 512, and 1024 respectively. More parameters and training methods of the GAN model can be found in [11].

Fig.6(a) shows the training process of the NICE model for the London smart meter data set. As the number of iterations increases, the loss function of the training set and the validation set decrease gradually. The loss function of the validation set is larger than that of the training set, which indicates that the NICE model occurred over-fitting. In addition, the loss function of the validation set is the smallest at the 272nd iterations. When the number of iterations is more than 272, the loss function of the training set is slowly decreasing and the loss function of the validation set is gradually increasing, which indicates that the applicability of the NICE model is deteriorating. Therefore, the NICE model for London smart meter data set takes the parameters of 272nd iterations. Similarly, Fig.6(b) shows the training process of the NICE model for the Spanish transmission service operator data set. The NICE model for this data set takes the parameters of 20th iterations, since the loss function of the validation set is the smallest at the 20th iterations.

**B. SCENARIO GENERATION**

After training the deep generative network, the 8000 random numbers obeying the Gaussian distribution are used as the input data of VAE, GAN and NICE model. All generative networks will obtain 8000 new daily load profiles.

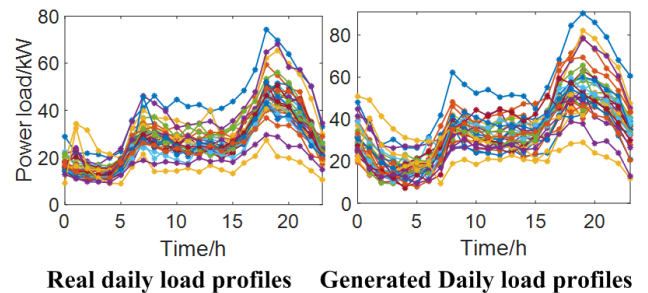


FIGURE 10. Generation of multiple daily load profiles.

Then, some random moments of the daily load profile are selected and the probability distributions at these moments are counted as shown in Fig.7.

Obviously, the overall trends of the probability density functions of the power loads generated by VAE and GAN are consistent with that of the real samples, but the specific details are quite different. For example, the GAN and VAE can't fit the peak and valley of probability density function well. In contrast, the probability density function of the power load generated by the NICE model is very similar to that of real samples, which indicates that the NICE model has a better ability to fit probability distribution than VAE and GAN.

As shown in Fig.8, the mean, standard deviation and wave rate are calculated to compare the volatility of the daily load profiles generated by the NICE model with that of the real load profiles.

The probability distributions of the volatility characteristics (e.g. mean, standard deviation and wave rate) of the generated daily load profiles by different deep generative



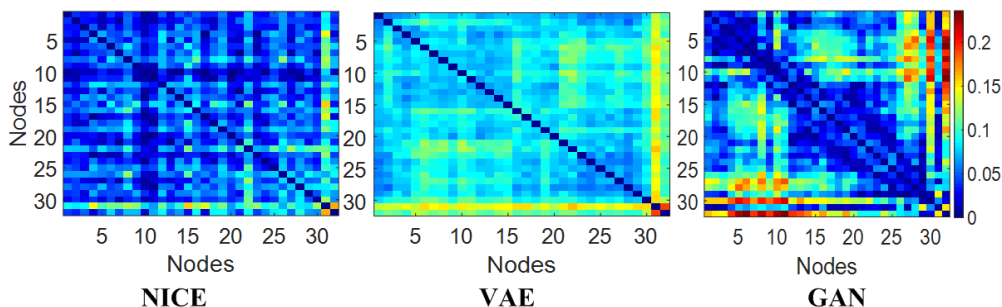


FIGURE 11. Error matrixes of different generative networks.

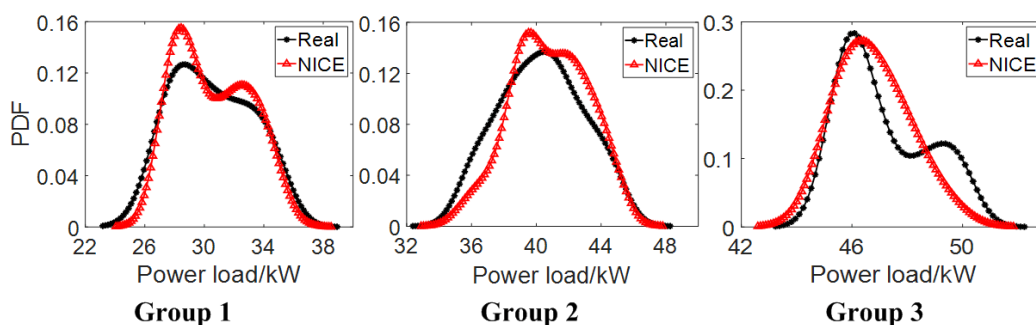


FIGURE 12. Probability density functions of different groups.

networks are almost the same as the real situation, which shows that these models can well simulate the volatility of real daily load profiles. Further, the performance of the NICE model is better than that of GAN and VAE, because the probability density function of the NICE model is closest to that of real samples.

**C. TEMPORAL CORRELATION**

Some daily load profiles generated by the NICE model are randomly selected in Fig.9 with comparison to some real samples from the test set. It is obvious that the daily load profiles generated by the NICE model closely resemble the real daily load profiles from the test set, which did not participate in the training process of the NICE model.

The trend of autocorrelation function between the generated daily load profile and the real daily load profile is the same, which shows that the NICE model can capture the temporal correlation of the real daily load profile well.

On the time axis of the load duration curve, the duration for which a certain load runs during the day is given. From the third row of Fig.9, it can be found that the load duration of the generated daily load profile and the real daily load profile almost coincide, which shows that they consume roughly the same power consumption in a day.

**D. SPATIAL CORRELATION**

The nodes of the distribution network have similar power consumption in one area, and are prone to simultaneous peaks or valleys. Therefore, it is necessary to consider the spatial

correlation when modeling daily load profiles. The London smart meter data set composed of 32 nodes is used to test the performance of each generative network.

Fig. 10 shows the real daily load profiles and the daily load profiles generated by the NICE model. It is obvious that the real daily load profiles of these 32 nodes have the same trend, which indicates that their power consumption is similar. The NICE model takes into account the spatial correlation among multiple nodes while generating the daily load profiles, which is in line with the actual situation.

Further, the spatial correlation among multiple daily load profiles is quantitatively analyzed using the Pearson coefficient. Firstly, a Pearson coefficient matrix among 32 nodes is calculated. Then, the Pearson coefficient matrixes of the generative networks are respectively subtracted from the Pearson coefficient matrix of the real samples to obtain error matrixes. Finally, the absolute value of the error matrix is obtained, and the visualized as shown in Fig. 11.

As can be seen from Fig. 11, most of the errors of the NICE model are between 0.05 and 0.1. The errors of VAE and GAN mostly fall between 0.1 and 0.2. The maximum error of VAE, GAN and NICE model does not exceed 0.194, 0.234 and 0.163 respectively. In general, the Pearson coefficient matrixes generated by these generative networks have a small difference from the Pearson coefficient matrix of the real sample, which indicates that they can well capture the spatial correlation of power load. Moreover, the NICE model performs better than VAE and GAN in capturing correlation.

## E. CONDITIONAL LOAD PROFILES GENERATION

In order to analyze the performance of the NICE model generating conditional daily load profiles, we calculate the mean of real samples. Then, the daily load profiles are divided into three groups based on the mean  $u(x)$ (kW):  $u(x) < 35$ ,  $35 \leq u(x) < 45$  and  $45 \leq u(x)$ . Further, every generated daily load profile is distributed to one group according to the mean value. The probability density function of real samples and generated daily load profiles in each group is shown in Fig. 12.

Obviously, the generated samples in each group and the real samples follow the same probability distribution. For example, in group 1, the mean values of the generated samples are relatively small, which belongs to the light load. While in group 3, the mean values of the generated load profiles are greater than 4.5, which can be utilized to simulate targeted heavy load days. These conditional load profiles play an important role in power system planning and operation. For example, relay protection devices may need to be tested under heavy and light loads to ensure that signals are properly signaled in the event of a fault. In short, the proposed method can generate daily load curves with special characteristics, which is of great help to distribution network planning.

## V. CONCLUSION AND FUTURE WORK

Modeling daily load profiles can help capture the uncertainty of power load, which is of great significance to the optimization and operation of the power system. In this paper, a flow-based generative network, the non-linear independent component estimation (NICE) is proposed to model daily load profiles. The proposed approach does not require explicitly estimate or fit the probability distribution of daily load profiles. The new data can be generated by sampling and training. Besides, it can be applied to model different loads in different regions only by adjusting the structure and parameters. After the simulation, the conclusions are as follows:

- 1) The proposed method utilizes historical data to model daily load profiles without artificially assuming the probability distribution of real samples. The simulation shows that the probability distribution of the daily load profiles generated by the NICE model is very similar to the probability distribution of the real samples, and the daily load profile generated by the NICE model has the same volatility characteristics as the real sample.
- 2) By comparing the real samples with the daily load profiles generated by the NICE model, it is found that the NICE model can accurately capture the spatial-temporal correlation of the real samples. In addition, the power consumption of generated daily load profiles closely resemble that of real samples. Simulation results show that the NICE model is more suitable for generating daily load profiles than VAE and GAN.
- 3) By simply classification, we can obtain the conditional daily load profiles with specific characteristics

(e.g. light load, medium load, and heavy load), which can be utilized to simulate targeted load days.

For future work, we can try to use the NICE model to model power profiles for renewable energy (e.g. wind and photovoltaic power). In this case, these power profiles can be used for power system optimization such as robust reactive power optimization. In addition, we can try to combine the NICE model with other generative networks (e.g. VAE and GAN) to improve performance.

## ACKNOWLEDGMENT

Thank you very much for the codes BoJone shared.

## REFERENCES

- [1] D. Lee and R. Baldick, "Load and wind power scenario generation through the generalized dynamic factor model," *IEEE Trans. Power Syst.*, vol. 32, no. 1, pp. 400–410, Jan. 2017.
- [2] L. Wenlong, S. Wang, L. Qi, and S. Xin, "Reactive power optimization of distribution network based on case-based reasoning," in *Proc. IEEE Power Energy Soc. Gen. Meeting (PESGM)*, Portland, OR, USA, Aug. 2018, pp. 1–5.
- [3] D. Yang, W. Liao, Y. Wang, K. Zeng, Q. Chen, and D. Li, "Data-driven optimization control for dynamic reconfiguration of distribution network," *Energies*, vol. 11, no. 10, p. 2628, 2018.
- [4] Z. Wang, C. Shen, F. Liu, and F. Gao, "Analytical expressions for joint distributions in probabilistic load flow," *IEEE Trans. Power Syst.*, vol. 32, no. 3, pp. 2473–2474, May 2017.
- [5] I. Goodfellow, "NIPS 2016 tutorial: Generative adversarial networks," 2017, *arXiv:1701.00160*. [Online]. Available: <http://arxiv.org/abs/1701.00160>
- [6] C.-H. Lee and C.-H. Wu, "A novel big data modeling method for improving driving range estimation of EVs," *IEEE Access*, vol. 3, pp. 1980–1993, 2015.
- [7] Z. Fotouhi, M. R. Hashemi, H. Narimani, and I. S. Bayram, "A general model for EV drivers' charging behavior," *IEEE Trans. Veh. Technol.*, vol. 68, no. 8, pp. 7368–7382, Aug. 2019.
- [8] X. Li, X. Zhang, L. Wu, P. Lu, and S. Zhang, "Transmission line overload risk assessment for power systems with wind and load-power generation correlation," *IEEE Trans. Smart Grid*, vol. 6, no. 3, pp. 1233–1242, May 2015.
- [9] S. Riaz, G. Verbic, and A. C. Chapman, "Computationally efficient market simulation tool for future grid scenario analysis," *IEEE Trans. Smart Grid*, vol. 10, no. 2, pp. 1405–1416, Mar. 2019.
- [10] C. Feng, F. Dabbene, and C. M. Lagoa, "A kinship function approach to robust and probabilistic optimization under polynomial uncertainty," *IEEE Trans. Autom. Control*, vol. 56, no. 7, pp. 1509–1523, Jul. 2011.
- [11] L. Gonog and Y. Zhou, "A review: Generative adversarial networks," in *Proc. 14th IEEE Conf. Ind. Electron. Appl. (ICIEA)*, Xi'an, China, Jun. 2019, pp. 1–6.
- [12] Y. Li, S. Tang, R. Zhang, Y. Zhang, J. Li, and S. Yan, "Asymmetric GAN for unpaired image-to-image translation," *IEEE Trans. Image Process.*, vol. 28, no. 12, pp. 5881–5896, Dec. 2019.
- [13] Y. Zhao, S. Takaki, H.-T. Luong, J. Yamagishi, D. Saito, and N. Minematsu, "Wasserstein GAN and waveform loss-based acoustic model training for multi-speaker text-to-speech synthesis systems using a WaveNet vocoder," *IEEE Access*, vol. 6, pp. 60478–60488, 2018.
- [14] Z. Guo, Y. Wan, and H. Ye, "A data imputation method for multivariate time series based on generative adversarial network," *Neurocomputing*, vol. 360, pp. 185–197, Sep. 2019.
- [15] Z. Chen and C. Jiang, "Building occupancy modeling using generative adversarial network," *Energy Buildings*, vol. 174, pp. 372–379, Sep. 2018.
- [16] G. Ping, J. Chen, T. Pan, and J. Pan, "Degradation feature extraction using multi-source monitoring data via logarithmic normal distribution based variational auto-encoder," *Comput. Ind.*, vol. 109, pp. 72–82, Aug. 2019.
- [17] R. Prenger, R. Valle, and B. Catanzaro, "WaveGlow: A flow-based generative network for speech synthesis," 2018, *arXiv:1811.00002*. [Online]. Available: <http://arxiv.org/abs/1811.00002>
- [18] Y. Mu, J. Wu, N. Jenkins, H. Jia, and C. Wang, "A spatial-temporal model for grid impact analysis of plug-in electric vehicles," *Appl. Energy*, vol. 114, pp. 456–465, Feb. 2014.

- [19] J. M. Sexauer, K. D. McBee, and K. A. Bloch, "Applications of probability model to analyze the effects of electric vehicle chargers on distribution transformers," *IEEE Trans. Power Syst.*, vol. 28, no. 2, pp. 847–854, May 2013.
- [20] D. Tang and P. Wang, "Probabilistic modeling of nodal charging demand based on spatial-temporal dynamics of moving electric vehicles," *IEEE Trans. Smart Grid*, vol. 7, no. 2, pp. 627–636, Mar. 2016.
- [21] Y. Xia, B. Hu, K. Xie, J. Tang, and H.-M. Tai, "An EV charging demand model for the distribution system using traffic property," *IEEE Access*, vol. 7, pp. 28089–28099, 2019.
- [22] D. Fischer, B. Stephen, A. Flunk, N. Kreifels, K. B. Lindberg, B. Wille-Haussmann, and E. H. Owens, "Modeling the effects of variable tariffs on domestic electric load profiles by use of occupant behavior sub-models," *IEEE Trans. Smart Grid*, vol. 8, no. 6, pp. 2685–2693, Nov. 2017.
- [23] Z. Pan, J. Wang, W. Liao, H. Chen, D. Yuan, W. Zhu, X. Fang, and Z. Zhu, "Data-driven EV load profiles generation using a variational auto-encoder," *Energies*, vol. 12, no. 5, p. 849, 2019.
- [24] H. Wei, Z. Hongxuan, D. Yu, W. Yiting, D. Ling, and X. Ming, "Short-term optimal operation of hydro-wind-solar hybrid system with improved generative adversarial networks," *Appl. Energy*, vol. 250, pp. 389–403, Sep. 2019.
- [25] Y. Chen, Y. Wang, D. Kirschen, and B. Zhang, "Model-free renewable scenario generation using generative adversarial networks," *IEEE Trans. Power Syst.*, vol. 33, no. 3, pp. 3265–3275, May 2018.
- [26] 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.
- [27] J. Lan, Q. Guo, and H. Sun, "Demand side data generating based on conditional generative adversarial networks," *Energy Procedia*, vol. 152, pp. 1188–1193, Oct. 2018.
- [28] D. P. Kingma and P. Dhariwal, "Glow: Generative flow with invertible 1x1 convolutions," 2018, *arXiv:1807.03039*. [Online]. Available: <http://arxiv.org/abs/1807.03039>
- [29] L. Dinh, J. Sohl-dickstein, and S. Bengio, "Density estimation using real NVP," in *Proc. Int. Conf. Learn. Represent. (ICLR)*, Toulon, France, Apr. 2017, pp. 1–32.
- [30] L. Zhang and B. Zhang, "Scenario forecasting of residential load profiles," *IEEE J. Sel. Areas Commun.*, vol. 38, no. 1, pp. 84–95, Jan. 2020.
- [31] R. Prenger, R. Valle, and B. Catanzaro, "Waveglow: A flow-based generative network for speech synthesis," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process. (ICASSP)*, Brighton, U.K., May 2019, pp. 3617–3621.
- [32] L. Dinh, D. Krueger, and Y. Bengio, "NICE: Non-linear independent components estimation," in *Proc. Int. Conf. Learn. Represent. (ICLR)*, San Diego, CA, USA, May 2015, pp. 1–13.
- [33] J. Hertz, A. Krogh, B. Lautrup, and T. Lehmann, "Nonlinear backpropagation: Doing backpropagation without derivatives of the activation function," *IEEE Trans. Neural Netw.*, vol. 8, no. 6, pp. 1321–1327, Nov. 1997.
- [34] Y. Yang, S.-P. Xiao, X.-S. Wang, and Y.-Z. Li, "Non-coherent radar detection probability for correlated gamma fluctuating targets in k distributed clutter," *IEEE Access*, vol. 6, pp. 3824–3827, 2018.
- [35] C. Yan, L. Fu, J. Zhang, and J. Wang, "A comprehensive survey on UAV communication channel modeling," *IEEE Access*, vol. 7, pp. 107769–107792, 2019.
- [36] J. Xue, R. Birke, L. Y. Chen, and E. Smirni, "Spatial-temporal prediction models for active ticket managing in data centers," *IEEE Trans. Netw. Service Manag.*, vol. 15, no. 1, pp. 39–52, Mar. 2018.

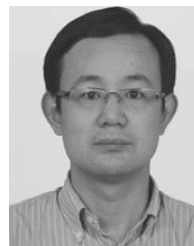


**LEIJIAO GE** (Member, IEEE) received the Ph.D. degree in electrical engineering from Tianjin University, Tianjin, China, in 2016. He is currently a Lecturer with the School of Electrical and Information Engineering, Tianjin University. His main research interests are smart distribution networks, cloud computing, and big data.



**WENLONG LIAO** (Graduate Student Member, IEEE) was born in Longyan, China. He received the B.S. degree in electrical engineering from China Agricultural University, Beijing, China, in 2017. He is currently pursuing the M.S degree in electrical engineering with Tianjin University, Tianjin, China.

His current research interests include smart grids, machine learning, and renewable energy.



**SHOUXIANG WANG** (Senior Member, IEEE) received the B.S. and M.S. degrees from the Shandong University of Technology, Jinan, China, in 1995 and 1998, respectively, and the Ph.D. degree from Tianjin University, Tianjin, China, in 2001, all in electrical engineering. He is currently a Professor with the School of Electrical and Information Engineering, and also the Deputy Director of the Key Laboratory of Smart Grid of Ministry of Education, Tianjin University. His

main research interests are distributed generation, microgrid, and smart distribution systems.



**BIRGITTE BAK-JENSEN** (Senior Member, IEEE) received the M.Sc. degree in electrical engineering and the Ph.D. degree in modeling of high voltage components from the Institute of Energy Technology, Aalborg University, Aalborg, Denmark, in 1986 and 1992, respectively.

From 1986 to 1988, she was with Electrolux Elmotor A/S, Aalborg, Denmark, as an Electrical Design Engineer. She is currently a Professor of intelligent control of the power distribution system with the Department of Energy Technology, Aalborg University. Her fields of interest are mainly related to the operation and control of the distribution grid including power quality and stability in power systems and taking integration of dispersed generation and smart grid issues like demand response into account. Also, the interaction between the electrical grid and the heating and transport sector is a key area of interest.



**JAYAKRISHNAN RADHAKRISHNA PILLAI** (Senior Member, IEEE) received the M.Tech. degree in power systems from the National Institute of Technology, Calicut, India, in 2005, the M.Sc. degree in sustainable energy systems from The University of Edinburgh, Edinburgh, U.K., in 2007, and the Ph.D. degree in power systems from Aalborg University, Aalborg, Denmark, in 2011.

He is currently an Associate Professor with the Department of Energy Technology, Aalborg University. His current research interests include distribution system analysis, grid integration of electric vehicles and distributed energy resources, smart grids, and intelligent energy systems.

...