IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received December 17, 2021, accepted December 27, 2021, date of publication January 11, 2022, date of current version January 21, 2022.

*Digital Object Identifier 10.1109/ACCESS.2022.3141875*

# Validation Methods for Energy Time Series Scenarios From Deep Generative Models

**EIKE CRAMER[1,2], LEONARDO RYDIN GORJÃO[3,4,5,6], ALEXANDER MITSOS[1,7,8], BENJAMIN SCHÄFER[9,10], DIRK WITTHAUT[3,4], AND MANUEL DAHMEN[1]**

[1]Forschungszentrum Jülich GmbH, Institute of Energy and Climate Research—Energy Systems Engineering (IEK-10), 52425 Jülich, Germany
[2]RWTH Aachen University, 52062 Aachen, Germany
[3]Forschungszentrum Jülich, Institute for Energy and Climate Research—Systems Analysis and Technology Evaluation (IEK-STE), 52428 Jülich, Germany
[4]Institute for Theoretical Physics, University of Cologne, 50937 Köln, Germany
[5]German Aerospace Center (DLR), Institute of Networked Energy Systems, 26129 Oldenburg, Germany
[6]Department of Computer Science, OsloMet—Oslo Metropolitan University, 0130 Oslo, Norway
[7]JARA-ENERGY, 52425 Jülich, Germany
[8]RWTH Aachen University—Process Systems Engineering (AVT.SVT), 52074 Aachen, Germany
[9]School of Mathematical Sciences, Queen Mary University of London, London E1 4NS, U.K.
[10]Faculty of Science and Technology, Norwegian University of Life Sciences, 1432 Ås, Norway

Corresponding author: Manuel Dahmen (m.dahmen@fz-juelich.de)

This work was supported in part by the Helmholtz School for Data Science in Life, Earth and Energy (HDS-LEE) through the Helmholtz Association of German Research Centres by the Uncertainty Quantification—From Data to Reliable Knowledge (UQ) under Grant ZT-I-0029; and in part by the European Union's Horizon 2020 Research and Innovation Program under the Marie Sklodowska-Curie under Grant 840825.

**ABSTRACT** The design and operation of modern energy systems are heavily influenced by time-dependent and uncertain parameters, e.g., renewable electricity generation, load-demand, and electricity prices. These are typically represented by a set of discrete realizations known as scenarios. A popular scenario generation approach uses deep generative models (DGM) that allow scenario generation without prior assumptions about the data distribution. However, the validation of generated scenarios is difficult, and a comprehensive discussion about appropriate validation methods is currently lacking. To start this discussion, we provide a critical assessment of the currently used validation methods in the energy scenario generation literature. In particular, we assess validation methods based on probability density, auto-correlation, and power spectral density. Furthermore, we propose using the multifractal detrended fluctuation analysis (MFDFA) as an additional validation method for non-trivial features like peaks, bursts, and plateaus. As representative examples, we train generative adversarial networks (GANs), Wasserstein GANs (WGANs), and variational autoencoders (VAEs) on two renewable power generation time series (photovoltaic and wind from Germany in 2013 to 2015) and an intra-day electricity price time series form the European Energy Exchange in 2017 to 2019. We apply the four validation methods to both the historical and the generated data and discuss the interpretation of validation results as well as common mistakes, pitfalls, and limitations of the validation methods. Our assessment shows that no single method sufficiently characterizes a scenario but ideally validation should include multiple methods and be interpreted carefully in the context of scenarios over short time periods.

**INDEX TERMS** Artificial neural networks, machine learning, time series analysis, uncertainty, stochastic processes, solar power generation, wind power generation.

## I. INTRODUCTION

The design and operation of modern energy systems are subject to uncertainties from many different sources [1]. Renewable electricity generation from wind turbines and solar PV panels depends on the weather and fluctuates on multiple time scales [2]. The integration of new consumers and different sectors increases both the complexity and the flexibility of the system on the demand side [3]. Electricity prices in liberalized markets show strong fluctuations reflecting the increasing variability of supply and demand [4], [5]. Balancing these fluctuations is a central challenge in the transition

The associate editor coordinating the review of this manuscript and approving it for publication was Mehul Raval.

to a future sustainable energy system. Hence, optimal design and operation of the energy system crucially depend on an accurate quantification and representation of the different stochastic processes [6]–[8]. For numerical optimization, the distributions of stochastic processes are often discretized in the form of scenarios. Each scenario then describes a possible realization over a given time period [6], [7], such that the (relatively small) collection of scenarios accurately represents the distribution [9], [10]. In most cases, energy time series exhibit highly correlated time steps and do not follow standard distributions. Thus, scenario generation requires specialized methods to sample from a stochastic process.

Recent research in scenario generation for energy time series focuses on deep generative models (DGMs), i.e., deep artificial neural networks (ANNs) that implicitly learn the distribution of a set of training data and allow for sampling without making assumptions about the underlying distribution [11]. Two prominent DGMs for scenario generation are (*i*) variational autoencoders (VAEs) [12] and (*ii*) generative adversarial networks (GANs) [13]. A popular modification of GANs are so-called Wasserstein GANs (WGANs), i.e., to use Wasserstein loss functions [14]. Applications of VAEs and (W)GANs include learning distributions of PV and wind power generation [15]–[22], concentrated solar power generation [23], electric vehicle power demand [24], and residential load [25].

Both VAEs and (W)GANs use unsupervised learning algorithms, and their loss functions do not explicitly enforce the generation of realistic data. Thus, to ensure a reliable representation of the stochastic processes, it is crucial to assess whether the generated data shares the same characteristics as the original data. For energy scenarios, this validation must rely on methods giving relevant insight into the essential characteristics of time series data. Furthermore, validation methods must be applied and their results must be interpreted correctly, in particular, for time series over short time periods, to ensure comparable and reliable results. While there is an ongoing discussion on validation of other types of data such as generated images [26], [27], we are not aware of a comprehensive discussion about the validation of DGM-generated energy time series scenarios.

To start this discussion, we critically assess the state of the art of application of validation methods and interpretation of their results in the DGM-based energy scenario generation literature. To this end, we review the application of the most commonly used validation methods for energy time series scenarios, namely probability density function (PDF), autocorrelation function (ACF), and power spectral density (PSD). For the analysis of more complex features like peaks, bursts, and plateaus, we propose using the multifractal detrended fluctuation analysis (MFDFA) as an additional validation method. Furthermore, we highlight shortcomings of the validation methods and explicate frequently encountered pitfalls in the interpretation of the validation results, in particular, in the context of scenarios over short time periods. Our assessment specifically focuses on the application of the validation methods in the context of independent and identically distributed (iid) DGM-generated scenarios. Note that other generation approaches, e.g., approaches that consider a day-ahead setting where there is additional information available, may require different validation methods. In our numerical experiments, we train GANs, WGANs, and VAEs on scenarios of three historical energy time series, i.e., PV and wind power generation as well as intra-day price. The validation results of the generated scenario are then compared to the historical data. We use the generated data to illustrate a proper application and interpretation of the validation methods and debate possible misconceptions about the information retrieved by the validation methods. We emphasize that our evaluation does not aim to assess whether the considered DGMs can reproduce the specific features of the data under consideration. Instead, we aim to assess whether the validation methods allow for the conclusions often made in the literature. While our assessment focuses on scenario generation in the field of energy systems, all considered validation methods apply to any iid time series scenario set in general and therefore can be applied in other domains as well.

Throughout this paper, we use the following notation: Scalars are denoted using light face (e.g., $x$) and vectors are denoted in boldface (e.g., $\mathbf{x}$). ANNs are denoted using boldface and capital letters (e.g., $\mathbf{G}$). Any values generated or inferred by an ANN are marked with a tilde (e.g., $\tilde{x}$). We use the subscript $\boldsymbol{\theta}$ for distribution functions that result from an ANN transformations (e.g., $p_{\boldsymbol{\theta}}(\tilde{\mathbf{x}})$). Stochastic processes are written as a function of time $x(t)$ and the corresponding time series is denoted using an index $x_t$. We use the term scenario to refer to one particular realization, e.g., a time series with a length of one day.

The remainder of this paper is organized as follows. Section II reviews the fundamental theory of GANs, WGANs, and VAEs, and briefly discusses their application in generating time series scenarios. In Section III, we review the validation methods and critically assess their application in the energy scenario generation literature. In Section IV, we apply the validation methods to generated scenarios and use the results to explicate common pitfalls and explain best practices. Finally, Section V provides a general discussion, raises open research questions, and concludes our work.

## II. DEEP GENERATIVE MODELS

In this section, we present the two most widely applied DGMs, namely VAEs and GANs with their modification to WGANs. In particular, we present the structural setup and the loss functions of the DGMs. For more comprehensive introductions, we refer to the original publications on GANs [13], WGANs [14], and VAEs [12].

### A. GENERATIVE ADVERSARIAL NETWORKS

GANs were proposed by Goodfellow *et al.* [13]. Their setup consists of two ANNs called generator $\mathbf{G}(\mathbf{z}; \boldsymbol{\theta}_{\boldsymbol{G}})$ and discriminator $\mathbf{D}(\mathbf{x}; \boldsymbol{\theta}_{\mathbf{D}})$ which are parameterized by the parameters

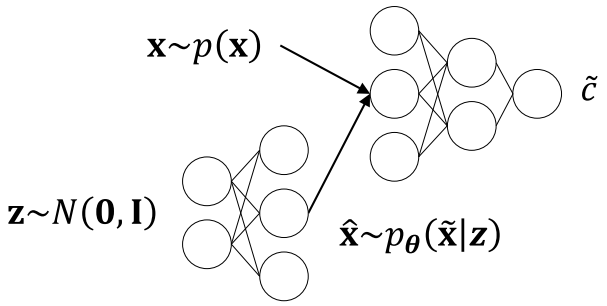


**FIGURE 1.** Generative adversarial network (GAN), with sampling distribution $\mathbf{z} \sim N(\mathbf{0}, \mathbf{I})$, true data distribution $\mathbf{x} \sim p(\mathbf{x})$, generated data distribution $\tilde{\mathbf{x}} \sim p_{\boldsymbol{\theta}}(\tilde{\mathbf{x}}|\mathbf{z})$, and the classification by the discriminator $\tilde{c}_{1/0}$.

$\boldsymbol{\theta}_\mathbf{G}$ and $\boldsymbol{\theta}_\mathbf{D}$, respectively. The generator is trained to transform samples of multivariate white noise $\mathbf{z} \sim N(\mathbf{0}, \mathbf{I})$, where $\mathbf{0}$ is a vector of zeros and $\mathbf{I}$ is the identity matrix, to data $\tilde{\mathbf{x}}$ that follows the distribution of the training data $\mathbf{x} \in X$ without learning the distribution $p_X(\mathbf{x})$ explicitly:

$$\tilde{\mathbf{x}} = \mathbf{G}(\mathbf{z}; \boldsymbol{\theta}_\mathbf{G}) \tag{1}$$

The generated data then follows the conditional distribution $\tilde{\mathbf{x}} \sim p_{\boldsymbol{\theta}}(\tilde{\mathbf{x}}|\mathbf{z})$, where we use the subscript $\boldsymbol{\theta}$ to indicate that the distribution is based on an ANN transformation. The discriminator acts as an adversary to the generator. It is trained to classify between real data $\mathbf{x}$ and generated data $\tilde{\mathbf{x}}$:

$$\tilde{c}_{1/0} = \mathbf{D}(\mathbf{x}; \boldsymbol{\theta}_\mathbf{D}) \tag{2}$$

In Equation (2), the output $\tilde{c}_{1/0}$ is the classification of the data where a value of 1 refers to real and 0 refers to fake samples. A sketch of the structural setup of a GAN is displayed in Figure 1.

Generator and discriminator are trained in an iterative loop. First, the generator samples new data which is then fed into the discriminator. The generator loss function is the negative expectation of the discriminator output, i.e., it quantifies the performance of the generator 'fooling' the discriminator:

$$\mathcal{L}_\mathbf{G}(\boldsymbol{\theta}_G) = -\mathbb{E}_Z\left[\mathbf{D}(\mathbf{G}(\mathbf{z}; \boldsymbol{\theta}_\mathbf{G}); \boldsymbol{\theta}_\mathbf{D})\right] \tag{3}$$

In Equation (3), $\mathbb{E}_Z$ is the expected value over $Z$. In the next part of the training loop, the discriminator is trained to distinguish between real and fake samples by means of supervised training. Its loss function is designed to give large values for false classifications and vice versa:

$$\mathcal{L}_\mathbf{D}(\boldsymbol{\theta}_D) = -\mathbb{E}_X\left[\mathbf{D}(\mathbf{x}; \boldsymbol{\theta}_\mathbf{D})\right] + \mathbb{E}_Z\left[\mathbf{D}(\mathbf{G}(\mathbf{z}; \boldsymbol{\theta}_\mathbf{G}); \boldsymbol{\theta}_\mathbf{D})\right] \tag{4}$$

Since the expression $\mathbb{E}_X\left[\mathbf{D}(\mathbf{x}; \boldsymbol{\theta}_\mathbf{D})\right]$ is not influenced by the generator training, Equations (3) and (4) can be combined to a min-max game also called adversarial training:

$$\min_{\boldsymbol{\theta}_\mathbf{G}} \max_{\boldsymbol{\theta}_\mathbf{D}} \quad \mathbb{E}_X\left[\mathbf{D}(\mathbf{x}; \boldsymbol{\theta}_\mathbf{D})\right] - \mathbb{E}_Z\left[\mathbf{D}(\mathbf{G}(\mathbf{z}; \boldsymbol{\theta}_\mathbf{G}); \boldsymbol{\theta}_\mathbf{D})\right] \tag{5}$$

The min-max game continues until the generator-discriminator pair reaches a Nash equilibrium, i.e., neither generator nor discriminator can improve without the other worsening. When the two networks are in Nash equilibrium, the generator is thought to reproduce the true distribution of the training data.

### B. WASSERSTEIN-GAN

In practice, standard GANs often converge to unstable Nash equilibria and show signs of mode dropping, i.e., a lack of diversity in the generated data [28], [29]. Furthermore, Arjovsky *et al.* [29] proved that the Nash equilibrium of the traditional training approach by Goodfellow *et al.* [13] does not represent the real target distribution.

To mitigate these issues, a popular modification to the GAN is to use the Wasserstein distance as a loss function [14]. It describes the least amount of work to transfer one distribution into another. In practice, the Wasserstein distance itself is never explicitly enforced. Instead, the WGAN relaxes the classification approach of the discriminator to a continuous output of a so-called critic:

$$\tilde{c} = \mathbf{C}(\mathbf{x}; \boldsymbol{\theta}_\mathbf{C}) \tag{6}$$

In Equation (6), $\mathbf{C}(\mathbf{x}; \boldsymbol{\theta}_\mathbf{C})$ is the critic with parameters $\boldsymbol{\theta}_\mathbf{C}$ and $\tilde{c}$ is a continuous confidence $\tilde{c} \in \mathbb{R}$ expressed by the critic on whether the data is real or fake.

Employing a critic with a real-valued output instead of a discriminator with a $\{0, 1\}$ classification avoids potential issues of vanishing gradients due to the sigmoid output activation of the discriminator and thereby leads to a more stable convergence [14]. To enforce the Wasserstein distance, the critic has to be restricted to be Lipschitz continuous. Arjovsky *et al.* [14] propose enforcing the Lipschitz constraint through weight clipping. However, other approaches using weight penalties have been proposed as well [30]. For more information about enforcing the Lipschitz constraint, the reader is referred to the original publication [14].

### C. VARIATIONAL AUTOENCODERS

VAEs were first proposed by Kingma and Welling [12]. The basic idea of VAEs is to extend the concept of autoencoders to a DGM. Autoencoders, in general, are (non-)linear transformations that compress the data to a lower-dimensional latent space and then rebuilt the data from the encoding. The parts of the autoencoder for encoding and decoding are called encoder and decoder, respectively, and are often built using ANNs:

$$\tilde{\mathbf{z}} = \mathbf{E}(\mathbf{x}; \boldsymbol{\theta}_\mathbf{E}) \tag{7}$$

$$\tilde{\mathbf{x}} = \mathbf{D}(\tilde{\mathbf{z}}; \boldsymbol{\theta}_\mathbf{D}) \tag{8}$$

Here, $\mathbf{x} \in X$ is the data, $\tilde{\mathbf{x}} \in \tilde{X}$ is the reconstructed data, $\tilde{\mathbf{z}} \in \tilde{Z}$ is the inferred latent space variable, and $\mathbf{E}(\mathbf{x}; \boldsymbol{\theta}_\mathbf{E})$ and $\mathbf{D}(\mathbf{z}; \boldsymbol{\theta}_\mathbf{D})$ are encoder and decoder networks with parameters $\boldsymbol{\theta}_\mathbf{E}$ and $\boldsymbol{\theta}_\mathbf{D}$, respectively. VAEs work similarly to regular autoencoders. However, their latent space is regularized to a multivariate Gaussian. Thereby, VAEs can generate new data by sampling from the known latent space distribution and transforming the samples with the decoder, similar to the GAN generator. A sketch of the structural setup of VAEs is shown in Figure 2.

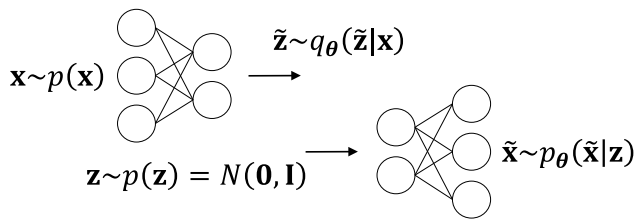


**FIGURE 2.** Variational autoencoder (VAE), with true data distribution $\mathbf{x} \sim p(\mathbf{x})$, generated data distribution $\tilde{\mathbf{x}} \sim p_{\boldsymbol{\theta}}(\tilde{\mathbf{x}}|\mathbf{z})$, sampling/latent distribution $p(\mathbf{z}) \sim N(\mathbf{0}, \mathbf{I})$, and inferred latent space distribution $q_{\boldsymbol{\theta}}(\tilde{\mathbf{z}}|\mathbf{x})$.

Ideally, VAEs would be trained via likelihood maximization. However, the nonlinear encoder and decoder function make the likelihood function intractable. Thus, a lower bound on the likelihood, also known as evidence lower-bound (ELBO), is used as the loss function:

$$\text{ELBO}(\boldsymbol{\theta}_{\mathbf{E}}, \boldsymbol{\theta}_{\mathbf{D}}) = \mathbb{E}_{\tilde{Z}}\left[\log\, p_{\boldsymbol{\theta}}(\tilde{\mathbf{x}}|\tilde{\mathbf{z}})\right] - D_{KL}\left[q_{\boldsymbol{\theta}}(\tilde{\mathbf{z}}|\mathbf{x})||p(\mathbf{z})\right] \tag{9}$$

Here, $p_{\boldsymbol{\theta}}(\tilde{\mathbf{x}}|\tilde{\mathbf{z}})$ is the distribution of the reconstructed data $\tilde{\mathbf{x}} = \mathbf{D}(\tilde{\mathbf{z}}; \boldsymbol{\theta}_{\mathbf{D}})$, i.e., the decoder output given a prior encoding $\tilde{\mathbf{z}} = \mathbf{E}(\mathbf{x}; \boldsymbol{\theta}_{\mathbf{E}})$, $D_{KL}[\cdot||\cdot]$ is the Kullback-Leibler divergence, a measure of similarity between distributions, $q_{\boldsymbol{\theta}}(\tilde{\mathbf{z}}|\mathbf{x})$ is the latent space distribution given the data $\mathbf{x}$, and $p(\mathbf{z}) = N(\mathbf{0}, \mathbf{I})$ is a multivariate standard Gaussian. The first term of the ELBO in Equation (9) maximizes the likelihood of the generated data. The second part with the Kullback-Leibler divergence enforces the standard Gaussian requirement for the latent space by minimizing the divergence between the inferred latent space distribution $q_{\boldsymbol{\theta}}(\tilde{\mathbf{z}}|\mathbf{x})$ and the target latent distribution $p(\mathbf{z}) = N(\mathbf{0}, \mathbf{I})$.

Since the initial publication by Kingma and Welling [12], extensions to the VAE concept were proposed that mostly concern the latent space distribution and propose Gaussian mixture models [31] and normalizing flows [32].

### D. DGM-BASED SCENARIO GENERATION

There are multiple works that propose different versions of the different DGMs reviewed above for energy scenario generation. Applications of GANs and WGANs include wind power generation [17], [19], PV power generation [16], [18], [20]–[22], and residential loads [25]. VAEs were applied to learn the distributions of PV and wind power generation [15], concentrated solar power [23], and electric vehicle power demand [24].

In all works on DGM-based scenario generation, the historical time series are cut into segments of equal length of typically one day. These segments, i.e., the historical scenarios, are then viewed as multidimensional iid data points, and the DGMs are trained to sample from the distribution of these data points.

## III. METHODS FOR VALIDATION OF ENERGY TIME SERIES SCENARIOS

In this section, we briefly review PDF, ACF, PSD, and MFDFA and explain their application as validation methods for DGM-generated scenarios. Furthermore, we critically assess the application of PDF, ACF, and PSD-based validation of DGM-generated energy scenarios in the literature.

### A. PROBABILITY DENSITY FUNCTION

The PDF describes the likelihood of a particular realization of a continuous random variable. For arbitrary data sets, there often exists no analytical distribution model and the PDF has to be estimated, e.g., by the non-parametric kernel density estimation (KDE) [33], [34]. For a finite set of univariate samples, KDE assigns a kernel function $K(\cdot)$, i.e., a non-negative function that integrates to 1, to the data. The density estimate is then given by the average over all $N$ samples:

$$\text{PDF}(x) = \frac{1}{hN}\sum_{i=1}^{N} K\left(\frac{x - x_i}{h}\right) \tag{10}$$

In Equation (10), $x_i$ are the samples, $x$ is the uncertain variable, and $h$ is a smoothing parameter called bandwidth. The most common kernel function is the Gaussian kernel [34]:

$$\text{PDF}(x) = \frac{1}{\sqrt{2\pi}hN}\sum_{i=1}^{N} \exp\left(-\frac{1}{2}\left(\frac{x - x_i}{h}\right)^2\right) \tag{11}$$

Equation (10) gives an explicit expression, which makes computing the PDF very efficient.

For DGM-generated scenarios, KDE can estimate either the overall PDF or the marginal PDF of the scenarios. For the PDF of all time steps, the time steps $x_t$ become the samples $x_i$ in Equation (10). The marginals are the sum over all the dimensions of a multivariate distribution, i.e., the sum over all time steps of a given scenario. Hence, the marginal PDF divided by the number of time steps describes the distribution of the daily mean.

In the literature on energy scenario generation, most authors evaluate the PDF on a linear scale [15], [17]–[19], [21]–[25], [35] or the integral over the PDF, i.e., the cumulative distribution function (CDF) [16], [21]. However, the linear scaled PDF and the CDF can only show differences between historical and generated scenarios for values of high likelihood. Thus, they hide potential mismatches in low-density regions, e.g., for electricity price distributions with heavy tails [36]. If the distributions are known to have heavy tails, the PDF should therefore also be investigated on a logarithmic scale to highlight the low-density regions.

Analyzing the marginal distributions reveals additional insight into the generated scenarios that cannot be extracted from the regular PDF, e.g., the full width of seasonal changes in PV or the distribution of high and low-price days. Still, most articles do not assess the marginal distribution, with the notable exception of the work in [37].

### B. AUTOCORRELATION FUNCTION

The ACF describes the internal correlation of a stochastic process $x(t)$ with a delayed version of itself $x(t - \tau)$. For discrete time series $x_t$, the ACF is a particular case of covariance describing the correlation between different time steps [38].

The auto-covariance $K_{xx}$ of a stationary time series is given by:

$$K_{xx}(\tau) = \mathbb{E}\left[(x_t - \mu_x)(x_{t+\tau} - \mu_x)\right] \tag{12}$$

In Equation (12), $\mu_x$ is the mean of the time series $x_t$ and $\tau$ the time-lag between the time steps. For the ACF, we consider the normalized auto-covariance, also known as the Pearson correlation

$$R_{xx}(\tau) = \frac{\mathbb{E}\left[(x_t - \mu_x)(x_{t+\tau} - \mu_x)\right]}{\sigma_t^2} \quad \in [-1, 1], \tag{13}$$

where $\sigma_t^2$ is the variance of the time series. The ACF has at least one maximum at $R_{xx}(\tau = 0) = 1$, where the expected value coalesces to the variance. Note that different normalizations for the auto-correlation function exist. However, in this work we refer to the Pearson correlation, Equation (13), as the ACF.

For the validation of iid DGM-generated scenarios, it is important to note that a concatenation of multiple scenarios would result in potentially unrealistic sequences. Therefore, ACF can only evaluate single scenarios, and an analysis of the entire data set is not possible. The limitation to short single scenarios also means that the ACF can be computed very efficiently. The literature approach most often used randomly selects a set of 3 to 4 historical scenarios and then searches the generated scenario set for the scenarios with the most similar ACF using the Euclidean distance as a metric, i.e., the ACF trajectory is viewed as a multidimensional point and the generated scenario with the ACF closest to the ACF of the historical scenario is selected as representative of the entire scenario set. Most authors then go on to interpret a good match of the ACFs to indicate a correct match of the temporal behavior by the DGM-generated scenarios [16]–[19], [21], [25], [37].

Opposed to this general understanding, we advise that the ACF results should be interpreted with caution. First, the typical comparison of ACFs of 3 to 4 historical scenarios to the best matching generated scenarios does not give any general information about the full generated scenario set since most of the scenarios are excluded from the evaluation. On the contrary, potential outliers that may show very different ACFs are systematically excluded by the best-match comparison. Second, a match of ACFs does not prove that two time series are the same or even similar, e.g., two time series can stem from different stochastic processes and still show the same ACF (c.f. Section IV-C). In conclusion, we find that the ACF analysis approach widely used in the literature can result in potentially spurious implications about the generated data and should therefore be treated with caution.

### C. POWER SPECTRAL DENSITY

Stochastic processes often exhibit periodic behavior, which can be analyzed in the frequency domain using the PSD. The PSD describes the distribution of the fluctuational power over a range of frequencies [38]. Here, the term power refers to the square of the signal and is distinct from the unit of the stochastic process, e.g., renewable power generation. The general form of the PSD $S_{xx}$, based on the Fourier transform $\hat{x}(f)$, is given by:

$$S_{xx}(f) = \lim_{T\to\infty} \frac{1}{T} |\hat{x}_T(f)|^2 \tag{14}$$

Here, $f$ is the frequency, $|\cdot|$ denotes the absolute value, and $\hat{x}_T(f)$ is the Fourier transform over a variable interval $T$ selected through an indicator function $I_T$, $\hat{x}_T(t) = \hat{x}(t) \cdot I_T$. The Fourier transform is given by:

$$\hat{x}(f) = \int_{-\infty}^{\infty} x(t) \exp(-2\pi jft) \mathrm{d}t \tag{15}$$

In Equation (15), $j$ is the imaginary unit. For practical applications, the PSD is computed using the fast Fourier transformation [39], which allows for very efficient computation. For a more detailed introduction, we refer to the literature [38]. The PSD of real-world time series is often noisy and difficult to read. Therefore, it is common to employ smoothing functions such as Welch's method [40], which computes the PSD over a set of overlapping segments instead of treating the time series as a whole. The average over these segments then gives a smooth estimate of the PSD.

The PSD is designed for consecutive time series. For the application of PSD to DGM-generated scenarios, all scenarios are concatenated to form a single time series. Hence, the periods relevant to the validation are between the Nyquist frequency, i.e., the shortest period supported by the PSD [38], and the scenario length. The PSD of any period longer than the scenario length describes the concatenation of scenarios, i.e., its use is inconsequential to the aim of validation. However, most authors present the PSD of longer periods [16], [18], [23]. For instance, the authors in [16] generate 24 h scenarios in 5 min resolution, but present the PSD over periods between 6 d and 1 h, which neglects short periods reflecting the short-term behavior. Therefore, we emphasize that the presented periods must reflect the scenario length and not truncate the short-term behavior.

### D. MULTIFRACTAL DETRENDED FLUCTUATION ANALYSIS

ACF and PSD form the basis of scenario validation in many contributions. However, the information they extract is limited, i.e., ACF can only quantify the correlations between time steps within a time series and PSD uncovers power-law decays of the spectrum. However, the time series data might include nontrivial features like peaks, bursts, and plateaus that are not represented in the ACF and PSD analysis, but still should be validated in order to gain confidence in the generated scenarios. To retrieve these features, we propose to use multifractal analysis.

Multifractal analysis aims to uncover the fractal composition of a stochastic process $x(t)$, i.e., the change in the fluctuation behavior relative to the sampling rate and the considered interval [41]. The term fractal refers to a process where the whole behaves similarly to parts of itself and is therefore also known as a self-similar process. A stochastic

process with constant fluctuation behavior, e.g., Brownian motion, consists of a single fractal and is therefore called monofractal [41]. If the fluctuation behavior changes with the sampling rate and/or the considered interval, the process is a composite of multiple fractals and hence called multifractal. For the fractal analysis, the stochastic process has to be dissected into its different scales. There are multiple algorithms to analyze the fractality of time series, e.g., wavelet analysis [42]–[44] and MFDFA [45]–[48]. We refer to reference [41] for a review of these methods. For our discussion, we focus on MFDFA, due to its straightforward application to discrete time series, and utilize a modification to moving windows to analyze short time series [47], [48]. In the following, we give a brief introduction to the MFDFA algorithm. For a more detailed explanation, we refer to the original MFDFA paper [46].

In a first step, the MFDFA algorithm computes the centered integral $Y_i$ over the finite time series $x_k$, i.e., the cumulative sum subtracted by the mean $\mu_x$, for all time steps $i$:

$$Y_i = \sum_{k=1}^{i} (x_k - \mu_x), \quad \forall\, i = 1, 2, \ldots, N \tag{16}$$

Next, the centered integral is split into $v = 1, 2, \ldots, N_s$ segments with a length of $s$ steps and a polynomial $y_v$ of order $m$ is fitted to each segment. Figure 3 shows an example with first and second-order polynomials. The segmentation and polynomial fitting is repeated for a spectrum of segment lengths $s \in \{m+2, \ldots, S\}$. Hence, the number of segments $N_s$ alters with the segment length. In this work, $S$ is selected to be the number of steps in a scenario. To quantify the variability of the detrended process, MFDFA then computes the variance function $F^2(v, s)$ of the data around the polynomial fit for a given segment $v$ of length $s$:

$$F^2(v, s) = \frac{1}{s} \sum_{i=1}^{s} [Y_{(v-1)s+i} - y_{v,i}]^2, \quad \forall\, v = 1, 2, \ldots, N_s \tag{17}$$

In Equation (17), $y_{v,i}$ denotes the polynomial value at the $i$-th point of the segment. The variance function is indicated as the shaded areas around the polynomials in Figure 3. Lastly, the MFDFA computes the $q$-th order fluctuation function $F_q(s)$ of the variance function $F^2(v, s)$ over the $N_s$ windows $v$:

$$F_q(s) = \left\{ \frac{1}{N_s} \sum_{v=1}^{N_s} [F^2(v, s)]^{q/2} \right\}^{1/q} \tag{18}$$

MFDFA aims to retrieve the fluctuation power by determining the power scaling of the fluctuation function

$$F_q(s) \sim s^{h(q)}, \tag{19}$$

where the exponent $h(q)$ is also known as the generalized Hurst coefficient [46]. To investigate $h(q)$, the fluctuation function $F_q(s)$ is displayed over the segment length $s$ in double logarithmic scaling, where $h(q)$ is the slope of $F_q(s)$. Parallel curves for different $q$-powers, i.e., constant Hurst coefficients $h(q)$, indicate a monofractal time series. A change in the slope for different $q$-powers indicates multifractal behavior. In general, low $q$-powers describe the mean fluctuation, and high $q$-powers extract bursts and peaks. Negative $q$-powers analyze plateaus and extended periods of low fluctuations.

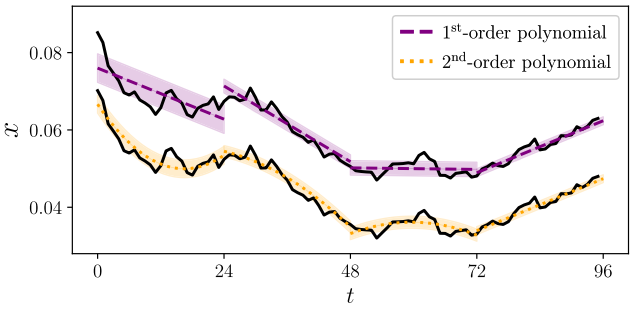


**FIGURE 3.** Wind capacity factor time series [-] with four non-overlaping windows of size $s = 24$ with 1st-order polynomial detrending (top) and 2nd-order polynomial detrending (bottom). The standard deviation of the detrended data is shown in the shaded areas, as given by Equation (17).

To our knowledge, there is no instance of fractal analysis in the energy scenario generation literature. Therefore, we propose the following approach for MFDFA of DGM-generated scenarios: Similar to PSD, MFDFA is designed to analyze consecutive time series. Therefore, the generated scenarios are concatenated for the evaluation. Then, we compute the fluctuation function using linear detrending with three positive and three negative $q$-powers. The original MFDFA [46] computes the variance over non-overlapping segments. For the analysis of shorter time series like the DGM-generated scenarios, we adopt the sliding window approach [47], [48]. Here, the window of length $s$ progresses step-wise, rendering a much higher number of segments $N_s$ for the analysis. Due to the concatenation, segments overlapping multiple scenarios are likely to connect unrealistic sequences and yield spurious results. Therefore, segment lengths exceeding half the scenario length have to be evaluated cautiously.

As shown in Equation (17), computing the MFDFA requires fitting polynomials, which makes the computation more expensive than PDF, ACF, and PSD. In particular, using the sliding window approach leads to a high number of fitting tasks.

### E. COMMONALITIES AND DIFFERENCES

Finally, we summarize the commonalities and differences between the four considered methods in Table 1. For instance, ACF and PSD check very similar categories with the only major difference that PSD analyzes the entire dataset. Importantly, only MFDFA and the PDF on logarithmic scaling analyze rare events. As neither of the two is typically used, this aspect is currently missing from the DGM-based scenario generation literature. Except for MFDFA, all methods are computationally very efficient and can be computed within milliseconds on a personal computer. MFDFA computation takes up to a minute for the considered datasets and the

**TABLE 1. Summary of commonalities and differences of PDF, ACF, PSD, and MFDFA.**

| | PDF | ACF | PSD | MFDFA |
|---|---|---|---|---|
| Probability distribution | ✓ | ✗ | ✗ | ✗ |
| Temporal dependencies | ✗ | ✓ | ✓ | ✓ |
| Full data set | ✓ | ✗ | ✓ | ✓ |
| Rare events | ✓[1] | ✗ | ✗ | ✓ |
| Nontrivial features | ✗ | ✗ | ✗ | ✓ |
| Easy interpretation | ✓ | ✓ | ✓ | ✗ |
| Require concatenation | ✗ | ✗ | ✓ | ✓ |
| Computationally efficient | ✓ | ✓ | ✓ | ✗ |
| Quantitative measure | ✗ | ✗ | ✗ | ✗ |

[1] Only on logarithmic scaling.

extended computational time is mostly due to the sliding window approach. Finally, the last line of Table 1 highlights that all validation methods discussed in this paper require visual interpretations of results. These visual interpretations are potentially subjective and we hypothesize that the lack of quantitative measures contributes to the often spurious conclusions often seen in the literature.

## IV. NUMERICAL STUDIES

In this section, we apply the previously discussed validation methods to DGM-generated scenarios of PV and wind capacity factors and intraday prices. To provide a variety of different scenario sets, we utilize GAN-, WGAN-, and VAE-generated scenarios in comparison to historical data. Nevertheless, our analysis is focused solely on the assessment of the validation methods and refrains from any implications about either of the three considered DGM. We start by describing the considered data sets and the model structures used for the DGMs. Next, we apply step-by-step PDF, ACF, PSD, and MFDFA to the historical and the DGM-generated time series scenarios.

### A. DGM TRAINING

To generate scenarios for our evaluation, we train GANs, WGANs, and VAEs on three different historical energy time series data sets. The data sets are:

- Daily total PV power generation in Germany from 2013 to 2015 [49],
- Daily total wind power generation in Germany from 2013 to 2015 [49],
- Intra-day price data from 2017 to 2019 from the European Energy Exchange AG (EEX), based in Leipzig, Germany, processed by the Fraunhofer Institute for Solar Energy Systems [50].

As described in Section II-D, the historical time series is cut into daily segments. Each segment is then viewed as a vector-valued iid sample of a multivariate distribution, and each time step is represented as a dimension [16]. The time series considered here are recorded in 15 min intervals, rendering scenarios comprised of 96 time-steps. The collection of all scenarios then is a set of 96 dimensional iid samples.

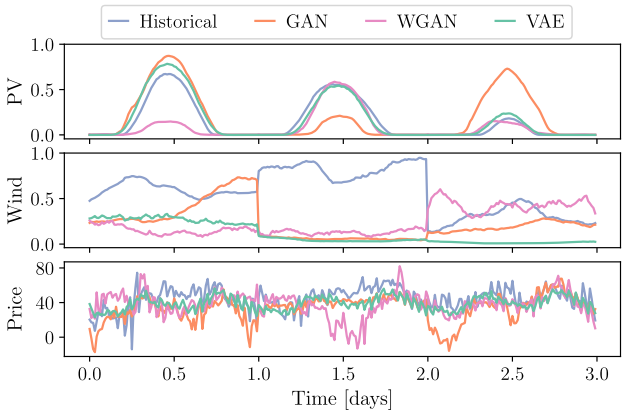


**FIGURE 4. Examples of PV capacity factor [-], wind capacity factor [-] from Germany in 2013 to 2015, and intra-day price [EUR/kWh] time series in the European Energy Exchange in 2017 to 2019 (labels shortened in figure). Time series are concatenations of three randomly selected scenarios from historical [49], [50] and GAN, WGAN, and VAE-generated scenario sets.**

None of the three historical time series contains any missing or faulty time steps. The intra-day price time series contains some extreme peaks. These peaks are important parts of the process and should not be considered outliers. PV and wind scenarios are scaled by the total installed capacity at each time step, resulting in the so-called capacity factor. All data sets are scaled to $[-1, 1]$ for GAN and WGAN and to $[0, 1]$ for the VAE to fit the respective 'tanh' and 'sigmoid' output activation functions. All DGMs are implemented using the open-source, Python-based machine learning library Tensorflow, version 2.4.0 [51]. Details of the respective DGM structures are listed in Table 2. Note that our work places a special focus on the correct application and interpretation of the validation methods as opposed to focusing on generating particularly good results in the evaluation, and the design of our DGMs is not optimized via rigorous hyperparameter optimization.

We train GAN and VAE using the Adam optimizer [54] and WGAN using RMSprop [55] as recommended in the WGAN paper [14]. All networks are trained for 2000 epochs with a learning rate of $10^{-5}$. As the loss functions of neither (W)GAN nor VAE provide stopping criteria for the training, we take to following approach to selecting the trained models: After every epoch we save the models, draw 1000 samples, and evaluate their PDF and PSD, see Sections III-A and III-C. After the 2000 epochs, we select the model from the epoch where the two metrics best match the historical data as judged through visual inspection. In Figure 4, we show some exemplary data of the three different historical time series [49], [50] and the generated scenarios. The time series are displayed over three days and are concatenations of randomly selected scenarios each.

Next, we apply the validation methods discussed in Section III to the generated scenarios.

### B. PROBABILITY DENSITY FUNCTION

We apply KDE with Gaussian kernels to estimate the PDF of the historical time series and the scenarios

**TABLE 2.** DGMs: ANN layer setup for the WGAN, GAN, and VAE. 1D Conv are convolutional layers [52] and 1D ConvT are transposed convolutional (deconvolutional) layers [53]. The attributes for different layers are: Fully connected: (Number of neurons), reshape: (output dim 1, output dim 2, ...), 1D Convolution: (number of filters, filter size, strides, padding), and 1D Convolution Transpose: (number of filters, filter size, strides, padding).

| WGAN | | | GAN | | | VAE | | |
|---|---|---|---|---|---|---|---|---|
| generator | | | generator | | | decoder | | |
| **Layer** | **attributes** | **activation** | **Layer** | **attributes** | **activation** | **Layer** | **attributes** | **activation** |
| Fully connected | (1152) | ReLU | Fully connected | (1152) | ReLU | Fully connected | (1152) | ReLU |
| reshape | (96, 12) | ReLU | reshape | (96, 12) | ReLU | reshape | (96, 12) | ReLU |
| 1D ConvT | (12, 3, 1, 1) | ReLU | 1D ConvT | (12, 3, 1, 1) | ReLU | 1D ConvT | (12, 3, 1, 1) | ReLU |
| 1D ConvT | (1, 3 ,1, 1) | None | 1D ConvT | (1, 3, 1, 1) | None | 1D ConvT | (1, 3, 1, 1) | sigmoid |
| critic | | | discriminator | | | encoder | | |
| **Layer** | **attributes** | **activation** | **Layer** | **attributes** | **activation** | **Layer** | **attributes** | **activation** |
| 1D Conv | (12, 3, 1, 1) | LeakyReLU | 1D Conv | (12, 3, 1, 1) | ReLU | 1D Conv | (16, 5, 1, 0) | ReLU |
| 1D Conv | (4, 3, 1, 1) | LeakyReLU | 1D Conv | (4, 3, 1, 1) | ReLU | Flatten | (-) | None |
| Flatten | (-) | None | Flatten | (-) | None | Fully connected | (80) | ReLU |
| Fully connected | (1) | None | Fully connected | (1) | tanh | Fully connected | (20+20) | None |

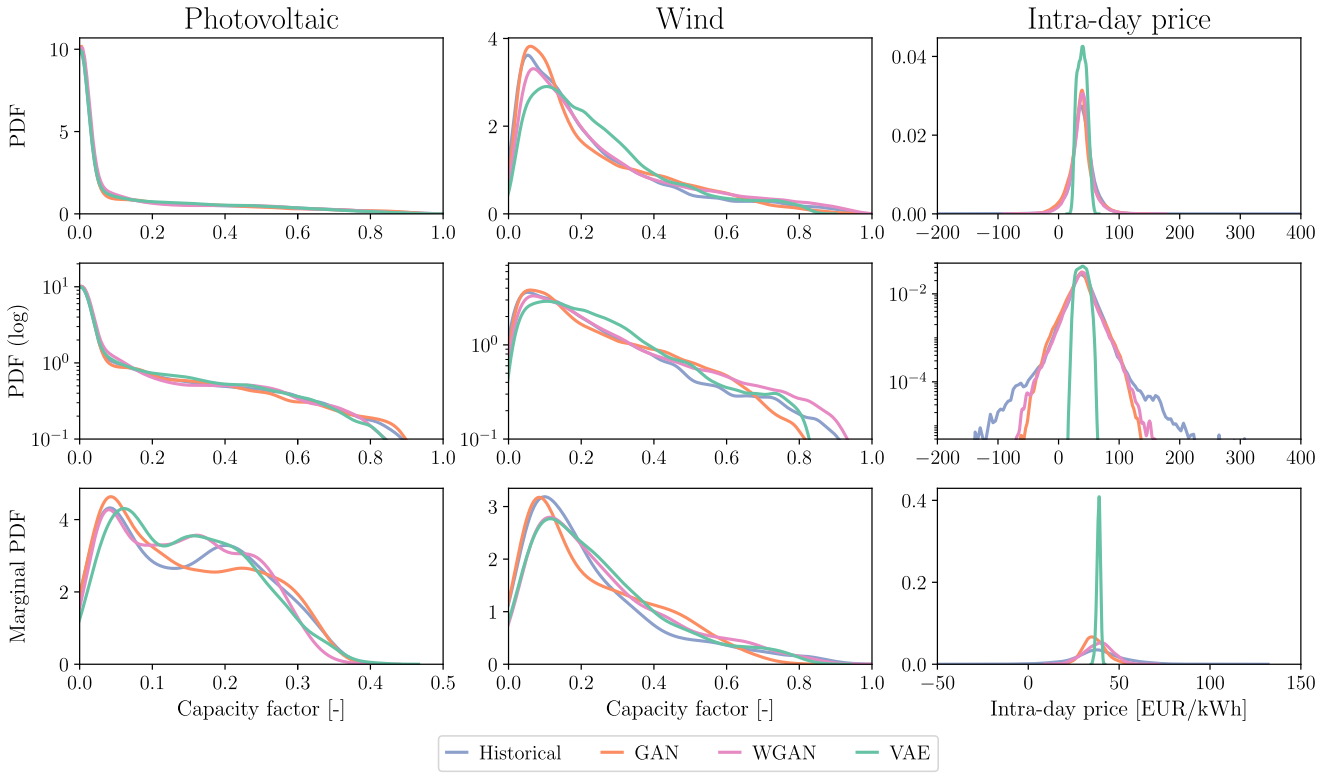


**FIGURE 5.** Kernel density estimates of the PDF of PV and wind capacity factor, and intra-day price scenarios. Historical datasets [49], [50] are compared to and GAN-, WGAN-, and VAE-generated scenarios.

generated using GAN, WGAN, and VAE. For completeness, we show the PDFs on linear and logarithmic scales and examine the scaled marginal distributions, i.e., the daily mean of capacity factor and intra-day price. All PDF calculations and plots are generated using the Python-based library Seaborn, version 0.11.1 [56], and Matplotlib, version 3.3.4 [57]. Figure 5 shows the results for PV and wind capacity factors, and intra-day prices, respectively.

On the linear scale, the GAN and WGAN-generated scenarios do not show any considerable divergence from the historical PDFs. Meanwhile, clear differences can be observed for the wind capacity factor and the intra-day price scenarios generated by the VAE. Thus, the PDF identifies the shifted distribution of VAE-generated wind capacity factor scenarios and the compressed distribution of VAE-generated intra-day price scenarios. Furthermore, the logarithmic scaling of the intra-day price PDF reveals that the GAN and WGAN fail to reproduce the heavy tails of the PDFs, i.e., they do not reproduce the rare price peaks that are present in the historical intra-day price time series. For both PV and wind capacity factors, the analysis of the PDFs in logarithmic scaling does not offer any additional insight into the distributions. Since most authors exclusively evaluate the linear scale, we emphasize the importance of analyzing the PDF on both linear and

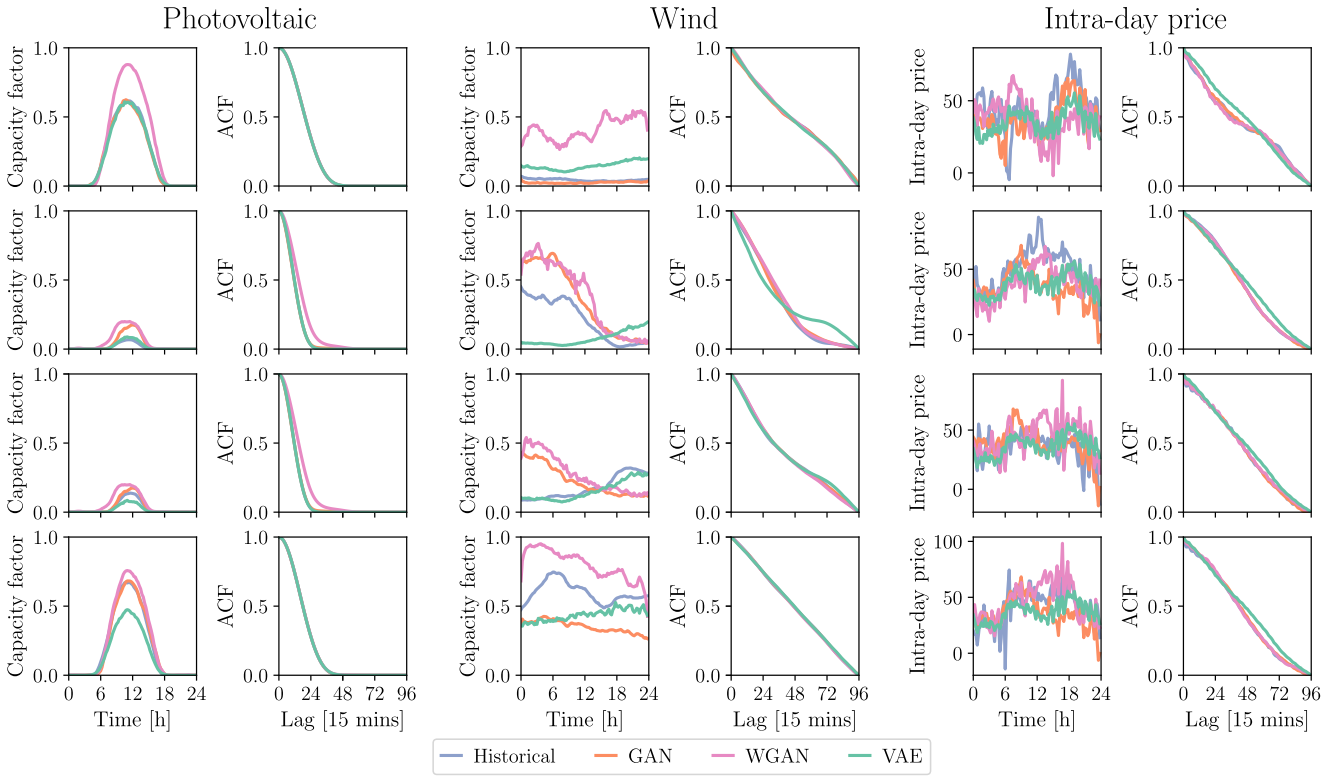


**FIGURE 6.** Autocorrelation function of selected historical scenarios and generated scenarios with Euclidean distance matching. Four exemplary scenarios of PV and wind capacity factor [-], and intra-day price [EUR/kWh] scenarios. Historical scenarios [49], [50] are compared to GAN-, WGAN-, and VAE-generated scenarios.

logarithmic scales, in particular, since the extreme events located in the heavy tails can critically affect system stability. PV and wind capacity factor distributions have finite support and no heavy tails. Hence, it is sufficient to consider the linear scale.

The marginal PDFs of the generated data show worse matches than the full PDFs, even for the distributions without heavy tails. For instance, all DGMs learn poor fits of the bimodal structure of the marginal PV capacity factor PDF, i.e., the marginal distributions are not represented well although the full distribution indicates a good match. The marginals of the VAE-generated intra-day price scenarios also display a much narrower distribution than the historical scenarios, which indicates that the VAE does not reproduce the full range of days with high and low prices. Note that this result is different from the also narrow intra-day price PDF, as it assesses the range of days with overall high or low prices as opposed to only the peaks.

### C. AUTOCORRELATION FUNCTION

For the analysis of the ACF, we follow the standard approach used in the literature of matching historical and generated scenarios by the best match of the ACF, see Section III-B. Figure 6 shows four selected scenarios and their respective ACFs for PV capacity factor, wind capacity factor, and intra-day prices, respectively.

For all three data sets, the Euclidean distance selection described in Section III-B finds DGM-generated scenarios with an ACF similar to the ACF of the respective historical scenario, indicating a good match of the autocorrelation of the respective sample. However, as discussed in Section III-B, the matching ACFs are no proof of a good representation of the historical time series. Indeed, Figure 6 shows that there are scenarios in the generated sets that exhibit similar autocorrelation as the 4 considered historical scenarios. Besides, similar ACFs do not necessarily indicate that two scenarios stem from the same stochastic process, e.g., the first row of wind and the second row of price scenario in Figure 6 have very similar ACFs despite describing two different data sets. In conclusion, we find that the ACF should only be tested if a matching autocorrelation is a specific requirement. In particular, the limitation to single scenarios and the resulting lack of subsequent implications about the full scenario sets are highly problematic. In any case, the results should be treated with caution.

### D. POWER SPECTRAL DENSITY

Following the standard literature approach, we concatenate the generated scenarios to form a single continuous time series and then calculate the PSD using Welch's method [16], [18], [23]. The results for PV capacity factor, wind capacity factor, and intra-day prices are shown in Figure 7. We observe

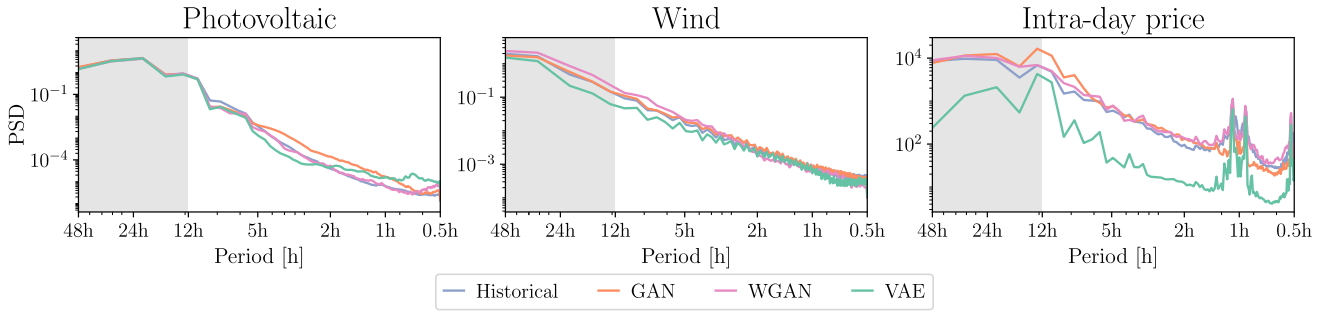


**FIGURE 7.** Power spectral density from Welch's method [40] of PV and wind capacity factor, and intra-day price scenarios. Historical datasets [49], [50] are compared to GAN-, WGAN-, and VAE-generated scenarios. Periods with potentially overlapping scenarios marked by gray areas.

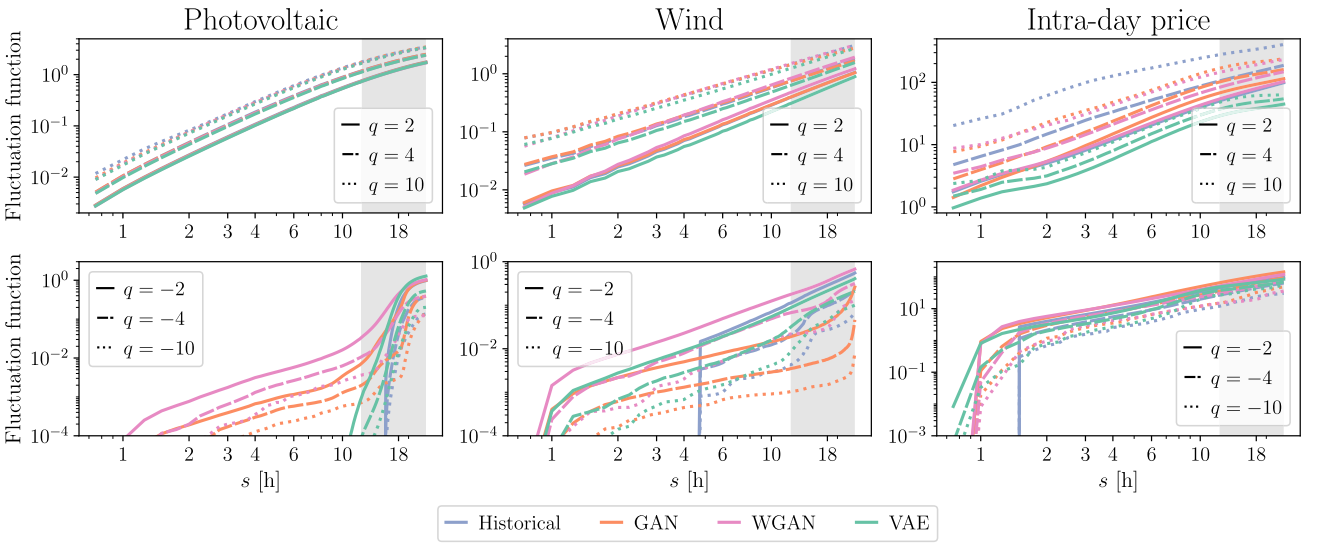


**FIGURE 8.** MFDFA of PV and wind capacity factor, and intra-day prices. Shown is the fluctuations function $F_q^2(s)$ with positive powers $q = 2$, $q = 4$, and $q = 10$ (upper row) and negative powers $q = -2$, $q = -4$, and $q = -10$ (lower row). Historical data [49], [50] is compared to GAN, WGAN, and VAE-generated scenarios. Segment length with potentially overlapping scenarios marked by gray area.

that the scenarios generated by the WGAN exhibit similar PSDs as the historical scenarios for all three data sets. The GAN- and VAE-generated scenarios show slight mismatches for PV and wind capacity factors, and there is a larger mismatch for the VAE-generated intra-day price scenarios. Specifically, the PSDs for the PV capacity factors reveal that the VAE-generated scenarios underestimate the fluctuations of medium length periods between 8 h and 2 h, and result in higher fluctuation for shorter time series, i.e., noisy scenarios. Meanwhile, the GAN-generated PV capacity factor scenarios have constantly higher fluctuations compared to the historical data. None of the above insights can be retrieved by relying on PDF and ACF alone.

We present the PSD over periods between 48 h and 30 min, where 30 min is the Nyquist frequency for the 15 min sampling interval. We mark the periods over 12 h in gray to indicate that they are likely to overlap different scenarios and lead to spurious results. Shifting the considered periods of our data proportional to [16] as discussed in Section III-C would result in periods between 3 weeks and 5 h. Such a shift disregards, for instance, the peaks in the intra-day price PSD (see Figure 7). Therefore, we conclude that it is indispensable to constrain the PSD to relevant periods, especially when working with concatenated time series.

### E. MULTIFRACTAL DETRENDED FLUCTUATION ANALYSIS

We compute the fluctuation function over concatenated scenarios using linear detrending and $q$-powers $q = 2$, $q = 4$, $q = 10$, and $q = -2$, $q = -4$, $q = -10$, respectively. The code for the MFDFA is available on GitHub [58]. Figure 8 shows the results for PV and wind capacity factor, and intra-day price scenarios. We mark in gray segment lengths $s$ greater or equal to half the scenario length that are likely to overlap multiple scenarios and potentially result in spurious results. For the positive $q$-powers, all the DGM-generated scenarios exhibit virtually no differences in the fluctuation function compared to the historical PV time series. For the wind capacity factor, the generated scenarios match the multifractal behavior of the historical time series, indicated by the change in slope for different $q$-powers. However, the

fluctuation functions for WGAN with $q = 2$ and $q = 4$ show steep slopes that indicate stronger fluctuation in the long-term behavior, i.e., more rapid changes in the time series compared to the historical data. In the DGM scenario generation literature, the ACF is used to retrieve such deviations in the autocorrelation. However, in our application, the ACFs of the 4 analyzed scenarios do not indicate a false autocorrelation. Only MFDFA is able to achieve this task among the here considered validation methods. For the intra-day price, the fluctuation functions of the DGM-generated scenarios deviate strongly from the historical time series, except for the $q = 2$ fluctuation function of the GAN and WGAN scenarios. While the PSDs of GAN and WGAN indicate a good representation of the fluctuations in the intra-day price data, the MFDFA reveals that the DGMs capture only the average fluctuations, i.e., the PSD fails to retrieve the missing bursts and peaks in the generated scenarios.

The MFDFA analysis with negative powers shows a steep drop-off by the fluctuation function for all three historical time series due to the data precision of two decimals for PV and wind capacity factor and three decimals for the intraday price. None of the DGM-generated scenarios match the drop-off and instead show fluctuation functions as a result of the untruncated values. In the case of the PV scenarios, the drop-off can also be attributed to the constant period of zero production during night times. Here, the higher fluctuations indicate some noise in the data that was already shown by the PSD, see Section IV-D. The analysis of wind scenarios with negative $q$-powers over longer segments highlights that WGAN-generated scenarios have high long-term, low amplitude noise around the plateaus. Contrarily, GAN-generated scenarios display lower long-term noise than the historical data. We argue that peaks, bursts, and plateaus are essential elements of the process that need to be considered. Restricting the analysis to PDF, ACF, and PSD disregards these non-trivial elements and can lead to false conclusions about the generated scenarios.

## V. CONCLUSION AND FUTURE WORK

### A. SUMMARY

We discussed commonly used validation methods for DGM-generated scenarios and their application in the literature with a special focus on energy time series. Our assessment focused on commonly used validation methods based on the probability density function (PDF), autocorrelation function (ACF), and power spectral density (PSD). In addition, we proposed using multifractal detrended fluctuation analysis (MFDFA) as a validation method for more complex features of time series like peaks, bursts, and plateaus. Our assessment is specifically concerned with the validation of identically and independently distributed (iid) scenario sets and discusses the different validation methods in this particular context.

Each of the four validation methods analyzes a particular feature of the generated data. For instance, the PDF describes the likelihood of the time series taking a certain value, while ACF, PSD, and MFDFA consider the relation of time steps in their order of occurrence. The literature approach of analyzing ACF only considers 3-4 single scenarios and thereby fails to give insight into the full DGM-generated scenario set. Furthermore, there is no guarantee that scenarios with matching ACF stem from the same stochastic process. Thus, the ACF results must be treated with caution. In contrast, PDF, PSD, and MFDFA analyze the entire data set and thereby allow to derive conclusions about DGMs ability to generate realistic scenarios.

Between PDF, PSD, and MFDFA there is no one exclusive validation method to analyze the generated scenarios. For instance, the PDF only considers the data distribution and neglects the correlation of time steps. The PSD retrieves insight into fluctuation and periodicity but considers neither the distribution nor the different scales of the stochastic process. The MFDFA neither assesses the distribution of the data nor the periodic behavior. However, MFDFA is uniquely able to dissect the different scales of the time series and retrieve, e.g., the missing bursts and peaks in the generated intra-day price scenarios. Analyzing the data using PDF, ACF, and PSD disregards these non-trivial elements, which can lead to false conclusions about the generated scenarios. To summarise, no method precludes using other methods as well, and only the complement of multiple methods allows for an assessment of the generated data from different perspectives.

### B. PRACTICAL IMPLICATIONS

In practice, requirements from the envisaged application of the scenarios should be used to select a set of appropriate validation methods. For instance, the PDF ranks most relevant if the application requires the scenarios to represent the full range of possible realizations. If the scenarios have to include the extreme peaks of, e.g., the intra-day price, only MFDFA can give the relevant insight. Some validation methods can be omitted if the data is known to not include the analyzed features, e.g., it appears unnecessary to show the logarithmic scale PDF of the PV data considered in this work as the distribution does not have heavy tails. Regardless of the downstream application, we stress that all validation methods must be applied and interpreted correctly, e.g., considering the short time period characteristic of scenarios. For instance, PSD and MFDFA are based on concatenated scenarios and should not be evaluated for intervals longer than the scenario length. Similarly, PDFs with heavy tails should be evaluated in logarithmic scaling in addition to the standard linear scale. In general, we do not expect the computational complexity of any of the validation methods to be prohibitive as they are typically applied only once after the training of the DGMs. In any case, PDF and PSD can be computed very efficiently using kernel density estimation and Welch's method, respectively, and the ACF only has to be computed for very few time steps. MFDFA is more complex compared to the other methods, however, it can be computed in reasonable times.

### C. OPEN RESEARCH QUESTIONS

We stress that the list of validation approaches presented here is non-exhaustive, and other not included tests may offer valuable insight as well and should be selected depending on the requirements of the application. Furthermore, there are many open research questions in the field of scenario validation.

Importantly, PDF, ACF, PSD, and MFDFA are all based on visual inspection of their respective outputs, which makes their interpretation subjective and potentially error-prone. This is further highlighted by our discussion and we argue that naive interpretation of the validation results can be susceptible to spurious conclusions. Consequently, future research on scenario validation should focus on deriving quantitative metrics that allow for an easier and comparable interpretation. A representative set of such quantitative metrics could then result in a comprehensive taxonomy allowing for consistent and reproducible validation of generated scenario sets. Furthermore, quantitative metrics could be used to better design the DGMs, e.g., in hyperparameter tuning.

As emphasized throughout our paper, this work concerns only the validation of iid DGM-generated scenarios. However, scenarios obtained with other machine-learning models may require different validation approaches. For instance, autoregressive models [59] generate scenarios based on inputs, which makes each set of scenarios specific to those inputs and, thereby, nontrivial to compare.

Another important aspect of validation is multivariate analysis, i.e., the validation of scenarios that include multiple potentially correlated stochastic processes at the same time. The four validation methods discussed in this work are all inherently univariate, which makes their application to multivariate time series impossible. Therefore, multivariate analysis ranks highly important as there are strong correlations between different energy time series, e.g., renewable power generation and electricity prices [60].

Besides improved analysis of the data itself, the application of the generated scenarios must be considered. For instance, stochastic programming [6] is a highly relevant problem in many energy system circumstances. Here, a set of scenarios is used to discretize the true probability distribution of the stochastic processes, and different sets of selected scenarios should lead to consistent objective function values [9].

### D. CONCLUSION

In conclusion, validation methods for DGM-based scenario generation should receive more attention. In general, we find that relying on single validation methods is not sufficient and the generated scenarios should always be analyzed using multiple methods. Research on DGM-based scenario generation should emphasize accurate application and interpretation of methods that detail the relevant properties in the data they wish to reproduce. With the lack of loss functions that detail the quality of the generated data, scenario validation can be a valuable asset in the design and testing of DGM model structures. Furthermore, future development of new validation methods should focus on the application of scenarios in downstream decision-making, e.g., in numerical optimization.

## REFERENCES

[1] A. Mitsos, N. Asprion, C. A. Floudas, M. Bortz, M. Baldea, D. Bonvin, A. Caspari, and P. Schäfer, "Challenges in process optimization for new feedstocks and energy sources," *Comput. Chem. Eng.*, vol. 113, pp. 209–221, May 2018.

[2] M. Anvari, G. Lohmann, M. Wächter, P. Milan, E. Lorenz, D. Heinemann, M. R. R. Tabar, and J. Peinke, "Short term fluctuations of wind and solar power systems," *New J. Phys.*, vol. 18, no. 6, Jun. 2016, Art. no. 063027.

[3] T. Brown, D. Schlachtberger, A. Kies, S. Schramm, and M. Greiner, "Synergies of sector coupling and transmission reinforcement in a cost-optimised, highly renewable European energy system," *Energy*, vol. 160, pp. 720–739, Oct. 2018.

[4] G. Wolff and S. Feuerriegel, "Short-term dynamics of day-ahead and intraday electricity prices," *Int. J. Energy Sector Manage.*, vol. 11, no. 4, pp. 557–573, Sep. 2017.

[5] J. Märkle-Huß, S. Feuerriegel, and D. Neumann, "Contract durations in the electricity market: Causal impact of 15 min trading on the EPEX SPOT market," *Energy Econ.*, vol. 69, pp. 367–378, Jan. 2018.

[6] J. M. Morales, A. J. Conejo, H. Madsen, P. Pinson, and M. Zugno, *Integrating Renewables in Electricity Markets: Operational Problems*, vol. 205, 1st ed. New York, NY, USA: Springer, 2013.

[7] A. J. Conejo, M. Carrión, and J. M. Morales, *Decision Making Under Uncertainty in Electricity Markets*, vol. 1, 1st ed. Boston, MA, USA: Springer, 2010.

[8] P. Schäfer, H. G. Westerholt, A. M. Schweidtmann, S. Ilieva, and A. Mitsos, "Model-based bidding strategies on the primary balancing market for energy-intense processes," *Comput. Chem. Eng.*, vol. 120, pp. 4–14, Jan. 2019.

[9] M. Kaut and S. W. Wallace, "Evaluation of scenario-generation methods for stochastic programming," *Pacific J. Optim.*, vol. 4, no. 2, pp. 257–271, 2007, doi: 10.18452/8296.

[10] H. Heitsch and W. Römisch, "Scenario reduction algorithms in stochastic programming," *Comput. Optim. Appl.*, vol. 24, nos. 2–3, pp. 187–206, Feb. 2003.

[11] S. Bond-Taylor, A. Leach, Y. Long, and C. G. Willcocks, "Deep generative modelling: A comparative review of VAEs, GANs, normalizing flows, energy-based and autoregressive models," 2021, *arXiv:2103.04922*.

[12] D. P. Kingma and M. Welling, "Auto-encoding variational Bayes," 2013, *arXiv:1312.6114*.

[13] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in *Proc. 27th Int. Conf. Neural Inf. Process. Syst. (NIPS)*, vol. 2 Cambridge, MA, USA: MIT Press, 2014, pp. 2672–2680. [Online]. Available: https://proceedings.neurips.cc/paper/2014/file/5ca3e9b122f61f8f06494c97b1afccf3-Paper.pdf

[14] M. Arjovsky, S. Chintala, and L. Bottou, "Wasserstein GAN," 2017, *arXiv:1701.07875*.

[15] H. Zhanga, W. Hua, R. Yub, M. Tangb, and L. Dingc, "Optimized operation of cascade reservoirs considering complementary characteristics between wind and photovoltaic based on variational auto-encoder," in *Proc. MATEC Web Conf.*, vol. 246. Les Ulis, France: EDP Sciences, 2018, p. 01077.

[16] Y. Chen, Y. Wang, D. S. 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.

[17] C. Jiang, Y. Mao, Y. Chai, M. Yu, and S. Tao, "Scenario generation for wind power using improved generative adversarial networks," *IEEE Access*, vol. 6, pp. 62193–62203, 2018.

[18] 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.

[19] Y. Zhang, Q. Ai, F. Xiao, R. Hao, and T. Lu, "Typical wind power scenario generation for multiple wind farms using conditional improved Wasserstein generative adversarial network," *Int. J. Electr. Power Energy Syst.*, vol. 114, Jan. 2020, Art. no. 105388.

[20] Y. Chen, P. Li, and B. Zhang, "Bayesian renewables scenario generation via deep generative networks," in *Proc. 52nd Annu. Conf. Inf. Sci. Syst. (CISS)*, Mar. 2018, pp. 1–6.
[21] C. Jiang, Y. Chen, Y. Mao, Y. Chai, and M. Yu, "Forecasting spatio-temporal renewable scenarios: A deep generative approach," 2019, *arXiv:1903.05274*.
[22] J. Schreiber, M. Jessulat, and B. Sick, "Generative adversarial networks for operational scenario planning of renewable energy farms: A study on wind and photovoltaic," in *Proc. Int. Conf. Artif. Neural Netw.* Springer, 2019, pp. 550–564.
[23] Y. Qi, W. Hu, Y. Dong, Y. Fan, L. Dong, and M. Xiao, "Optimal configuration of concentrating solar power in multienergy power systems with an improved variational autoencoder," *Appl. Energy*, vol. 274, Sep. 2020, Art. no. 115124.
[24] 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, Mar. 2019.
[25] Y. Gu, Q. Chen, K. Liu, L. Xie, and C. Kang, "GAN-based model for residential load generation considering typical consumption patterns," in *Proc. IEEE Power Energy Soc. Innov. Smart Grid Technol. Conf. (ISGT)*, Feb. 2019, pp. 1–5.
[26] T. Salimans, I. Goodfellow, W. Zaremba, V. Cheung, A. Radford, and X. Chen, "Improved techniques for training GANs," in *Proc. 30th Int. Conf. Neural Inf. Process. Syst.* Red Hook, NY, USA: Curran Associates, 2016, pp. 2234–2242.
[27] A. Borji, "Pros and cons of GAN evaluation measures," *Comput. Vis. Image Understand.*, vol. 179, pp. 41–65, Feb. 2019.
[28] L. Metz, B. Poole, D. Pfau, and J. Sohl-Dickstein, "Unrolled generative adversarial networks," 2016, *arXiv:1611.02163*.
[29] M. Arjovsky and L. Bottou, "Towards principled methods for training generative adversarial networks," 2017, *arXiv:1701.04862*.
[30] I. Gulrajani, F. Ahmed, M. Arjovsky, V. Dumoulin, and A. Courville, "Improved training of Wasserstein GANs," in *Proc. 31st Int. Conf. Neural Inf. Process. Syst. (NIPS)*. Red Hook, NY, USA: Curran Associates, 2017, pp. 5769–5779.
[31] C. Guo, J. Zhou, H. Chen, N. Ying, J. Zhang, and D. Zhou, "Variational autoencoder with optimizing Gaussian mixture model priors," *IEEE Access*, vol. 8, pp. 43992–44005, 2020.
[32] D. Rezende and S. Mohamed, "Variational inference with normalizing flows," in *Proc. 32nd Int. Conf. Int. Conf. Mach. Learn.*, vol. 37, 2015, pp. 1530–1538. [Online]. Available: http://proceedings.mlr.press/v37/rezende15.pdf
[33] E. Parzen, "On estimation of a probability density function and mode," *Ann. Math. Statist.*, vol. 33, no. 3, pp. 1065–1076, Sep. 1962. [Online]. Available: https://www.jstor.org/stable/2237880
[34] R. A. Davis, K.-S. Lii, and D. N. Politis, "Remarks on some nonparametric estimates of a density function," in *Selected Works of Murray Rosenblatt*. New York, NY, USA: Springer, 2011, pp. 95–100.
[35] Z. Wang, C. Shen, and F. Liu, "A conditional model of wind power forecast errors and its application in scenario generation," *Appl. Energy*, vol. 212, pp. 771–785, Feb. 2018.
[36] B. Uniejewski, R. Weron, and F. Ziel, "Variance stabilizing transformations for electricity spot price forecasting," *IEEE Trans. Power Syst.*, vol. 33, no. 2, pp. 2219–2229, Mar. 2018.
[37] L. Ge, W. Liao, S. Wang, B. Bak-Jensen, and J. R. Pillai, "Modeling daily load profiles of distribution network for scenario generation using flow-based generative network," *IEEE Access*, vol. 8, pp. 77587–77597, 2020.
[38] P. Stoica and R. L. Moses, *Spectral Analysis of Signals*, 1st ed. Upper Saddle River, NJ, USA: Prentice-Hall, 2005.
[39] M. T. Heideman, D. H. Johnson, and C. S. Burrus, "Gauss and the history of the fast Fourier transform," *Arch. Hist. Exact Sci.*, vol. 34, no. 3, pp. 265–277, 1985.
[40] P. D. Welch, "The use of fast Fourier transform for the estimation of power spectra: A method based on time averaging over short, modified periodograms," *IEEE Trans. Audio Electroacoust.*, vol. AU-15, no. 2, pp. 70–73, Jun. 1967.
[41] H. Salat, R. Murcio, and E. Arcaute, "Multifractal methodology," *Phys. A, Stat. Mech. Appl.*, vol. 473, pp. 467–487, May 2017.
[42] J. F. Muzy, E. Bacry, and A. Arneodo, "Wavelets and multifractal formalism for singular signals: Application to turbulence data," *Phys. Rev. Lett.*, vol. 67, no. 25, pp. 3515–3518, Dec. 1991.
[43] E. Bacry, J. F. Muzy, and A. Arnéodo, "Singularity spectrum of fractal signals from wavelet analysis: Exact results," *J. Stat. Phys.*, vol. 70, nos. 3–4, pp. 635–674, Feb. 1993.
[44] J. F. Muzy, E. Bacry, and A. Arneodo, "The multifractal formalism revisited with wavelets," *Int. J. Bifurcation Chaos*, vol. 4, no. 2, pp. 245–302, Apr. 1994.
[45] J. W. Kantelhardt, E. Koscielny-Bunde, H. H. A. Rego, S. Havlin, and A. Bunde, "Detecting long-range correlations with detrended fluctuation analysis," *Phys. A, Stat. Mech. Appl.*, vol. 295, nos. 3–4, pp. 441–454, Jun. 2001.
[46] J. W. Kantelhardt, S. A. Zschiegner, E. Koscielny-Bunde, S. Havlin, A. Bunde, and H. E. Stanley, "Multifractal detrended fluctuation analysis of nonstationary time series," *Phys. A, Stat. Mech. Appl.*, vol. 316, nos. 1–4, pp. 87–114, Feb. 2002.
[47] Y. Zhou and Y. Leung, "Multifractal temporally weighted detrended fluctuation analysis and its application in the analysis of scaling behavior in temperature series," *J. Stat. Mech., Theory Exp.*, vol. 2010, no. 6, Jun. 2010, Art. no. P06021.
[48] X. Zhang, G. Zhang, L. Qiu, B. Zhang, Y. Sun, Z. Gui, and Q. Zhang, "A modified multifractal detrended fluctuation analysis (MFDFA) approach for multifractal analysis of precipitation in Dongting lake basin, China," *Water*, vol. 11, no. 5, p. 891, Apr. 2019.
[49] *Open Power Systems Data: Time Series*, Open Power Syst. Data, 2019. Accessed: Mar. 30, 2020, doi: 10.25832/time_series/2019-06-05.
[50] Energy-Charts. (2020). *Electricity Production and Spot Prices*. Accessed: Mar. 30, 2020. [Online]. Available: https://energy-charts.info/charts/price_spot_market/chart.htm?l=en&c=DE
[51] M. Abadi *et al.* (2015). *TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems*. [Online]. Available: https://www.tensorflow.org/
[52] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*, 1st ed. Cambridge, MA, USA: MIT Press, 2016.
[53] A. Radford, L. Metz, and S. Chintala, "Unsupervised representation learning with deep convolutional generative adversarial networks," 2015, *arXiv:1511.06434*.
[54] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," 2014, *arXiv:1412.6980*.
[55] T. Tieleman and G. Hinton, "Lecture 6.5–RMSprop: Divide the gradient by a running average of its recent magnitude," Neural Netw. Mach. Learn., Coursera, Mountain View, CA, USA, Lect. Slides 6.5, 2012, pp. 26–30. [Online]. Available: http://www.cs.toronto.edu/%7Etijmen/csc321/slides/lecture_slides_lec6.pdf
[56] M. L. Waskom, "Seaborn: Statistical data visualization," *J. Open Source Softw.*, vol. 6, no. 60, p. 3021, 2021.
[57] J. D. Hunter, "Matplotlib: A 2D graphics environment," *Comput. Sci. Eng.*, vol. 9, no. 3, pp. 90–95, May/Jun. 2007.
[58] L. R. Gorjão, G. Hassan, J. Kurths, and D. Witthaut, "MFDFA: Efficient multifractal detrended fluctuation analysis in Python," 2021, *arXiv:2104.10470*.
[59] S. I. Vagropoulos, E. G. Kardakos, C. K. Simoglou, A. G. Bakirtzis, and J. P. S. Catalão, "ANN-based scenario generation methodology for stochastic variables of electric power systems," *Electr. Power Syst. Res.*, vol. 134, pp. 9–18, May 2016.
[60] R. Weron, "Electricity price forecasting: A review of the state-of-the-art with a look into the future," *Int. J. Forecasting*, vol. 30, no. 4, pp. 1030–1081, 2014.

**EIKE CRAMER** received the B.Sc. and M.Sc. degrees in energy and process engineering from the Technical University of Berlin, in 2017 and 2019, respectively. He is currently pursuing the Ph.D. degree with the Jülich Research Center, Institute of Energy and Climate Research—Energy Systems Engineering (IEK-10), Germany, and the Helmholtz School for Data Science in Life, Earth, and Energy (HDS-LEE), Germany.

**LEONARDO RYDIN GORJÃO** received the B.Sc. degree in physics from the University of Lisbon, Portugal, in 2014, the M.Sc. degree in physics from the University of Bonn, Germany, in 2017, and the Ph.D. degree in physics from the University of Cologne, Germany, and the University of Oslo, Norway, in 2021. He is currently working with Oslo Metropolitan University, Norway, the DLR Institute of Networked Energy Systems, Germany, and IRCN, University of Tokyo, Japan.

**ALEXANDER MITSOS** received the Dipl.-Ing. degree in chemical engineering from the University of Karlsruhe, in 1999, and the Ph.D. degree in chemical engineering from MIT, in 2006. He is currently a Full Professor (W3) with RWTH Aachen University, Germany, and the Director of the Laboratory for Process Systems Engineering (AVT.SVT). He also has a joint appointment at Forschungszentrum Jülich, Germany, where he is the Director of IEK-10 Energy Systems Engineering.

**BENJAMIN SCHÄFER** received the Diploma degree in physics from the University of Magdeburg, Germany, in 2013, and the Ph.D. degree in physics from the University of Göttingen, in 2017. His research for the Ph.D. degree was carried out in Göttingen, Germany, London, U.K., and Tokyo, Japan. He worked as a Postdoctoral Researcher with the Max Planck Institute for Dynamics and Self-Organization, Göttingen, Technical University Dresden, Germany, and as a Marie Sklodowska-Curie Research Fellow with the Queen Mary University of London. In 2021 he worked as Associate Professor in Physics at the Norwegian University of Life Sciences. Since 2022, he is leading a Young Investigator Group on Data-Driven Analysis of Complex Systems for a Sustainable Future at the KIT, Germany.

**DIRK WITTHAUT** received the M.Sc. and Ph.D. degrees in physics from the Technical University of Kaiserslautern, Kaiserslautern, Germany, in 2004 and 2007, respectively. He worked as a Postdoctoral Researcher with the Niels Bohr Institute, Copenhagen, Denmark, and the Max Planck Institute for Dynamics and Self-Organization, Göttingen, Germany. He has been a Guest Lecturer with the Kigali Institute for Science and Technology, Rwanda. Since 2014, he has been leading a Research Group at Forschungszentrum Jülich, Germany. He is currently an Assistant Professor with the University of Cologne, Germany.

**MANUEL DAHMEN** received the Diploma degree (Dipl.-Ing.) in computational engineering science and the Ph.D. degree (Dr.-Ing.) from RWTH Aachen University, Germany, in 2010 and 2018, respectively. In 2017, he joined the Jülich Research Center, Institute of Energy and Climate Research—Energy Systems Engineering (IEK-10), Germany, where he currently heads the Optimization and Machine Learning and the Industrial Energy Systems Groups.