Generating Synthetic Power Grids Using Exponential Random Graph Models

Francesco Giacomarra[*]
Department of Mathematics, Informatics and Geosciences, University of Trieste, 34128 Trieste, Italy

Gianmarco Bet[†]
Department of Mathematics and Computer Science "Ulisse Dini," University of Florence, 50134 Florence, Italy

Alessandro Zocca[‡]
Department of Mathematics, Vrije Universiteit Amsterdam, 1081 HV Amsterdam, the Netherlands

(Received 22 November 2023; revised 9 April 2024; accepted 9 May 2024; published 6 June 2024)

Synthetic power grids enable real-world energy system simulations and are crucial for algorithm testing, resilience assessment, and policy formulation. We propose a novel method for the generation of synthetic transmission power grids using exponential random graph (ERG) models. Our two main contributions are (1) the formulation of an ERG model tailored specifically for capturing the topological nuances of power grids and (2) a general procedure for estimating the parameters of such a model conditioned on working with connected graphs. From a modeling perspective, we identify edge counts per bus type and k-triangles as crucial topological characteristics for synthetic power-grid generation. From a technical perspective, we develop a rigorous methodology to estimate the parameters of an ERG constrained to the space of connected graphs. The proposed model is flexible, easy to implement, and successfully captures the desired topological properties of power grids.

DOI: 10.1103/PRXEnergy.3.023005

I. INTRODUCTION

Power grids are fundamental infrastructures of our modern societies and economies. A thorough understanding of the principles that govern the formation of these networks is crucial to guarantee their reliability at all times. However, network operators release only limited information on transmission power grids due to security concerns. Therefore, there is a substantial lack of real high-quality data. To address this problem, over the past two decades, synthetic grid-generation approaches have been extensively investigated by the research community. The main challenge has been to develop models flexible enough to replicate the very heterogeneous nature and peculiar properties of real power grids.

We propose a novel approach for the generation of synthetic power grids based on exponential random graph (ERG) models. The main idea of an ERG is to consider a parametric probability density over the space of all graphs and tune its parameters to encode desirable topological properties as soft constraints. From a modeling perspective, ERGs have been quite successful due to their flexibility, since they offer a very tractable alternative to the problem of sampling from involved graph subspaces. To the best of our knowledge, this class of models has not yet been considered in the power-systems literature. This is probably due to the intrinsic difficulty of generating graphs that are both sparse and connected, both key topological properties of transmission power grids.

The main contributions of this paper are the following. First, we propose an ERG model that captures the main topological properties of real power grids. Second, we give a general procedure to estimate the parameters of a wide class of ERG models with constraints using an algorithm based on Markov-chain Monte Carlo (MCMC) with noisy parameters, which we prove to converge to the set of parameters that satisfies the constraints imposed by the ERG model defined before. We present the results that we have obtained with our procedure, showing that the proposed model is flexible and captures the properties of possibly very different power grids while also being simple, easy to implement, and theoretically grounded. We remark that the proposed methodology is rather general and, except for the choices of graph statistics, is, in fact, not specific to synthetic power-grid generation.

[*]Corresponding author: francesco.giacomarra@phd.units.it
[†]Corresponding author: gianmarco.bet@unifi.it
[‡]Corresponding author: a.zocca@vu.nl

Due to the aforementioned flexibility and the fact that it requires only one real observed grid to be implemented, our approach can be useful in numerous scenarios where a large number of realistic samples are needed, e.g., training machine-learning models. What distinguishes our formulation from most methods available in the literature is the fact that it is grounded in a well-studied theoretical framework for probabilistic modeling, allowing for a more insightful analysis of certain phenomena using tools from random graph theory. Since our proposed model results in a Boltzmann-type distribution, it could in principle be used in combination with other models (e.g., energy-based generative models [1]) to build a more complex formulation in a compositional fashion [2]. Furthermore, in Sec. VI, we propose as a future direction a way to use our model as a baseline to generate more complex synthetic networks by exploiting the idea of reassembling smaller topologies into larger ones [3–5].

In the past two decades, various proposals have been put forward to address the challenge of generating synthetic yet realistic power grids. Some major commonalities among these works can be identified, based on the approach used to tackle the problem. We will briefly review here some of the proposed methods for synthetic grid generation. For a more comprehensive survey on the topic, see Refs. [6,7].

Similarly to the approach that we propose here, there have been several attempts to use already existing graph models either to directly generate power-grid topologies or as building blocks for more complex procedures. Examples of models used are the "small-world model" introduced by Watts and Strogatz in Ref. [8] and refined by Wang *et al.* into the "RT-nested small-world model" to generate synthetic power grids [9–12]. Another model used is the Chung-Lu model [13–15], used as a building block for the generation procedures presented in Refs. [16,17]. Similarly, in Ref. [18], a variation of the generalized random graph model [19] is proposed as a generative model. A very recent paper proposes a topological approach to power-grid generation using a modified version of the Erdős-Rényi model [20]. In Ref. [21], the authors sample grid topologies using the so-called graphons, or graph functions [22], which can be viewed as limiting objects for *dense* random graphs.

Many researchers argue that the geographical attributes of the area and/or the geographical locations of the nodes cannot be disregarded while generating synthetic power grids. Consequently, several models proposed in the literature put particular emphasis on the *spatial embedding* of the synthetic grids. A straightforward way to generate spatially embedded grids that are both connected and sparse is to solve the "minimum spanning tree" (MST) problem [23] given the desired locations of the nodes and by assigning weights at each possible edge based on some distance-cost function. MSTs are often used as the first step in building the topology of several synthetic grid-generation procedures [24–28].

Other approaches in the literature to obtain spatially embedded synthetic grids rely on clustering of nodes based on the geographical properties of the locations of the nodes [29–31]. After cluster identification, different procedures have been proposed to obtain the topologies with the desired properties considering as distinct the edges that connect nodes within the same cluster and those that connect nodes belonging to different clusters.

In Refs. [3,4,32], the authors propose to view the power grids from the perspective of a "network of networks," i.e., analyzing separately the subgraphs with the same voltage level in the grid, which they call fragments, and then the interconnections of these subgraphs as a new graph itself. This hierarchical view has led to the development of the so-called *sustainable data-evolution technology* (SDET) tool to create open-access synthetic grid data sets [5]. Using this method, new synthetic topologies are generated by reassembling fragments of real grids that were previously anonymized (to avoid disclosure of sensitive information). The idea of developing an anonymization procedure rather than a completely new generation method is also discussed in Ref. [33], where the grid topologies of real grids are left unchanged and only the electrical parameters are randomized to obfuscate sensitive information.

Finally, Ref. [34] proposes a deep-learning method for the generation of synthetic power grid. To our knowledge, this is the first attempt to solve this kind of problem using deep-learning techniques. This is because, in general, these approaches require a large amount of input data to provide accurate results.

The paper is structured as follows. In Sec. II, we introduce the graph-theoretic framework and discuss the main topological properties of power grids. In Sec. III, we introduce the ERG model along with the proposed specifications. The estimation procedure and a theoretical explanation of its convergence are discussed in Sec. IV. The results obtained using our proposed model specifications are presented in Sec. V. Lastly, in Sec. VI, we offer final remarks and highlight potential avenues for future research.

II. TRANSMISSION POWER GRIDS AS COMPLEX NETWORKS

Power grids are interconnected networks that deliver electricity from producers to consumers, consisting of nodes called buses connected through links called power lines. We can distinguish two main types of power networks, namely, the distribution network and the transmission network. Distribution networks have shorter power lines (often referred to as distribution power lines) and serve the function of transporting electricity for short distances and low voltage levels. The transmission network is used to transport electricity over long distances, with

longer power lines operating at high voltage. Coherently with other similar works in the literature [5,9,16–18,24–26,29,31,35–38], we focus solely on the transmission network and in this section we will highlight how to model such a network and its properties in a graph-theoretic framework.

A. Model description and preliminaries

A high-voltage transmission network can be modeled as a simple undirected unweighted graph $G = (V, E)$, where the nodes $V = \{1, \ldots, |V|\}$ represent electrical buses and the edges $E \subset V \times V$ represent the transmission lines connecting them. We denote by $n := |V|$ the number of nodes and by $m := |E|$ the number of edges. Any simple graph $G = (V, E)$ can be equivalently described by its *adjacency matrix* $A = A(G) \in \{0, 1\}^{n \times n}$, which is the square-symmetric matrix defined as

$$A_{ij} = \begin{cases} 1, & \text{if } (i,j) \in E, \\ 0, & \text{otherwise.} \end{cases} \tag{1}$$

We define the graph distance $d(i,j)$ between any two nodes $i \neq j$ as the length of the shortest path (in hops) between i and j, with $d(i,i) = 0$ and $d(i,j) = \infty$ if there are no paths between i and j. If $d(i,j) < \infty$ for any pair of nodes $i \neq j$, then the graph is said to be *connected*. The *average path length* or *characteristic path length* is the average length of the shortest path between any two nodes in the graph, i.e.,

$$\langle \ell \rangle = \frac{1}{n(n-1)} \sum_{i,j \in V} d(i,j). \tag{2}$$

For every node $i \in V$, we define its *degree* $k_i = \deg(i) \in \mathbb{N}$ as the number of nodes adjacent to i in G. The degree k_i of node $i \in V$ can be recovered as the sum of the i th row of the adjacency matrix, namely, $k_i = \sum_{j=1}^{n} A_{ij}$. The *average node degree* $\langle k \rangle$ of the graph G is

$$\langle k \rangle := \frac{1}{n} \sum_{i=1}^{n} k_i = \frac{2m}{n}.$$

The *degree matrix* $D(G)$ of the graph G is the square-diagonal matrix defined as $D(G) = \mathrm{diag}(k_1, \ldots, k_n)$. Another equivalent matrix representation of the graph G is given by its *Laplacian matrix* $L = L(G) \in \mathbb{R}^{n \times n}$, which is the square-symmetric matrix defined as $L(G) := D(G) - A(G)$, or, equivalently, as

$$L_{i,j} := \begin{cases} k_i, & \text{if } i = j, \\ -1, & \text{if } (i,j) \in E, \\ 0, & \text{otherwise.} \end{cases} \tag{3}$$

The Laplacian matrix is a useful graph-theoretic tool and its spectrum can be linked to many properties of the corresponding graph (for a detailed overview, see Ref. [39]). In particular, the second-smallest eigenvalue λ_2 of the Laplacian matrix, known as *algebraic connectivity*, is closely related to the connectivity of the graph itself [40]. In particular, $\lambda_2 > 0$ if and only if G is a connected graph and its magnitude reflects how well connected G is. Spectral graph theory has been shown to be an excellent tool for understanding the redistribution of power flows after line contingencies [41–45] and thus for designing more robust network topologies [46,47].

A *triangle* is a clique of size 3, i.e., a subgraph consisting of three nodes that have an edge between each pair of nodes. The total number of triangles $t_1(G)$ in a graph G can be calculated using its adjacency matrix as

$$t_1(G) = \frac{1}{6} \sum_{i=1}^{n} \sum_{j=1}^{n} \sum_{l=1}^{n} A_{ij} A_{jl} A_{li}. \tag{4}$$

More generally, for each $k \geq 1$, one defines *k-triangles* as the subgraphs in which the k triangles share the same edge (for some examples, see Fig. 1). We denote the total number of k-triangles in a graph G as $t_k(G)$.

The *local clustering coefficient* C_i of node $i \in V$ is defined as the ratio between the number of triangles $\mathcal{T}_G(i)$ to which node i belongs and the maximum number of triangles $t_G(i)$ that could possibly exist between node i and its k_i neighbors. Denoting by $N_i \subseteq V$ the *neighborhood* of i, i.e., the collection of nodes adjacent to i in G, the local clustering coefficient can be computed as

$$\begin{aligned} C_i &= \frac{\mathcal{T}_G(i)}{t_G(i)} = \frac{|\{(j,k) \in E : j,k \in N_i\}|}{k_i(k_i-1)/2} \\ &= \begin{cases} \frac{1}{k_i(k_i-1)} \sum_{j=1}^{n} \sum_{l=1}^{n} A_{ij} A_{jl} A_{li}, & \text{if } k_i \geq 2, \\ 0, & \text{if } k_i < 2. \end{cases} \end{aligned} \tag{5}$$

The *average clustering coefficient* C is defined as the average of the local clustering coefficients, i.e.,

$$C = \frac{1}{n} \sum_{i=1}^{n} C_i. \tag{6}$$

B. Electrical properties

In this work, we focus purely on the topology of synthetic grid graphs and thus do not consider the electrical properties of the nodes or of the lines. We refer the interested reader to Ref. [48] and references therein. However, it is key to distinguish the nodes based on the function of the corresponding substation, as this information is used in the generative procedure. In the context of power networks, we generally distinguish three types of nodes:

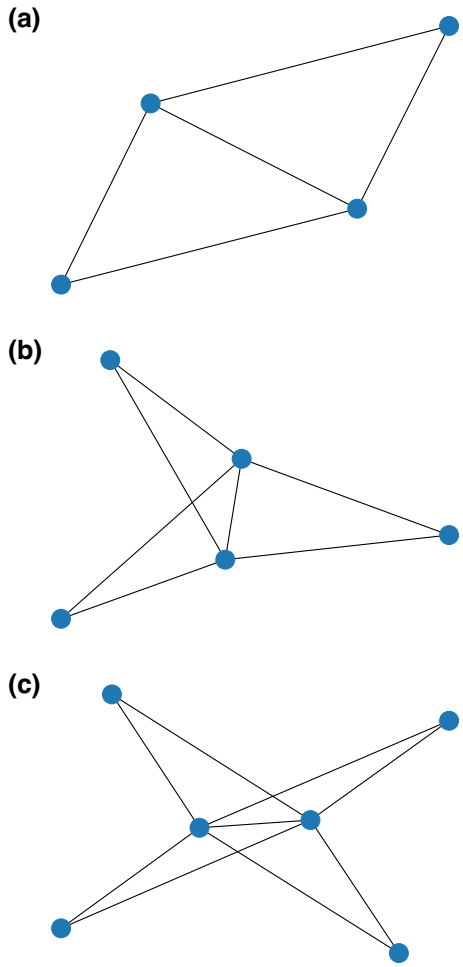

FIG. 1. Examples of (a) a 2-triangle, (b) a 3-triangle, and (c) a 4-triangle.

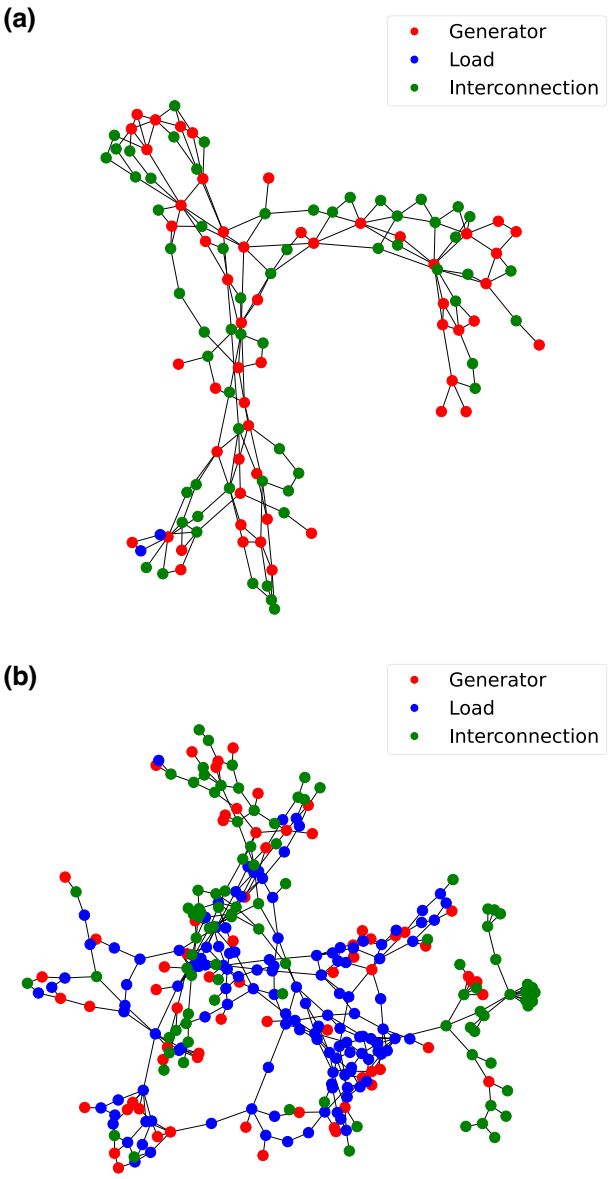

FIG. 2. The visualization of (a) the topology of the `118 IEEE` network and (b) the topology of the `300 IEEE` network. The colors of the nodes reflect the bus-type assignment. Since no geometric embedding has been given, the positions of the nodes have been determined using a force-directed graph-drawing algorithm [49].

(a) *generator nodes*, which represent the network components where the electricity is produced, e.g., conventional power plants, wind farms, or solar parks
(b) *load nodes*, which represent the network components where electricity is consumed, e.g., industrial districts, residential areas, or distribution network feeders
(c) *interconnections nodes*, which represent intermediate substations or transformers

From now on, we will refer to the subset of generator nodes in a power network with P, and similarly with L for the loads and with I for the interconnections.

In Fig. 2, we show the topologies of two grids made available in Ref. [50]. The two grids, namely, the `118 IEEE` and the `300 IEEE`, have 118 and 300 nodes, respectively. For each grid, in addition to the topology, we highlight the bus-type assignment. More details on the grids used in our work are given in Appendix A.

C. Topological properties of power grids

A large body of literature (see, e.g., Refs. [12,29,30,51]) has examined various topological properties of real power grids, which are instrumental in assessing the realism of synthetic grids. After an analysis of the available grids described and made available in Ref. [50], we now revisit and discuss some of these key properties. All these quantities for the considered grids are reported in Table V in Appendix A.

(a) *Connectivity.* Power grids are fully connected graphs, which means that there exists a path between any two nodes, except for rare emergency situations. This is due to the requirements for the reliability and security of the power-grid system. Indeed, under normal operating conditions, electrical power must be able to flow from any point on the grid to any other point. Most real power grids have an even stronger connectivity property, namely, they are *two-edge-connected graphs*. This property is, in fact, part of the standard contingency analysis known as $N-1$ *security*, which ensures that the power grid can withstand the failure of any single component (line, transformer, generator, etc.) without losing the ability to supply power to all the remaining loads.

(b) *Average node degree and sparsity.* It has been shown that the *average node degree* $\langle k\rangle$ in real power grids oscillates between 2 and 3 regardless of the network size [35,51]. Therefore, power grids are sparse graphs, which means that the number of edges is of the same order of magnitude as the number of nodes; informally, $|E| = \mathcal{O}(n)$. This can be intuitively explained by the high costs of building and maintaining transmission lines, as well as practical engineering constraints, such as avoiding transmission-line crossings.

(c) *Clustering coefficient and total number of triangles.* The average clustering coefficient C has been empirically observed to be much higher than that of other types of sparse graphs, which means that power grids tend to have many more triangles than other sparse graphs [12]. This could be due to the fact that the removal of a single line should not disconnect any node from the others and triangles are the simplest subgraph structure that allows for this property. We further investigate the clustering properties by looking at the number of k-triangles of the power grids in the publicly available data set [50]. Numerical evidence suggests that the number of triangles in power grids grows linearly with the number of edges (see Fig. 3). This result is consistent with the portion of literature claiming that the power networks follow the "small-world" property [8,12], which implies a higher clustering coefficient compared to other sparse random graphs with the same number of nodes (cf. Table V in Appendix A). On the other hand, the number of 2-triangles (and consequently that of any k-triangle with $k \geq 2$) is roughly constant—in fact, often very close to zero—and does not grow with the size of the network.

(d) *Bus-type assignment.* In Ref. [52], it is reported that in a typical power grid, 20–40% are generation buses, 40–60% are load buses, and about 20% are interconnection buses. The authors also suggest that there exists a correlation between bus-type assignment and several network-topology metrics. Our experimental analysis on the grids in Appendix A have found more heterogeneous values for the bus percentages; this might be due to the fact that the grids came from different sources and have different degrees of resolution (some buses might be aggregated together).

(e) *Average degrees per node type.* The degree distribution of the nodes in power grids has been shown to be different for the nodes of different types [9], motivating the introduction and the study of the average node degree per bus type $\langle k_P\rangle$, $\langle k_L\rangle$, $\langle k_I\rangle$, where

$$\langle k_a\rangle := \frac{1}{|a|}\sum_{i\in a} k_i, \quad a \in \{P, L, I\}. \tag{7}$$

(f) *Algebraic connectivity.* In Ref. [9] it has been said that algebraic connectivity scales as a power of the network size n, i.e., $\lambda_2 \sim n^p$, with p said to be in the range $[-1.376, -1.06]$. It is worth noting that our experimental analysis on the grids reported in Appendix A seems to suggest that the range or p should be widened but for a rigorous claim on the realistic values of λ_2 for power grids, a larger data set is needed.

(g) *Average shortest path length.* Assuming that power grids follow the "small-world" property as claimed by Albert and Barabasi [53] (but there is no consensus in the literature, cf. Ref. [51]), then the APL should grow proportionally to $\ln(m)/\ln(\langle k\rangle)$.

(h) *Graph diameter.* The graph diameter $d_{\max}$ of real power grids has been shown to scale roughly as $\sqrt{n}$ [3,16,31,54,55].

III. SYNTHETIC GRID GENERATION USING AN EXPONENTIAL RANDOM GRAPH MODEL

In this section, we briefly discuss the probabilistic details of the ERG model and the rationale behind the proposed model formulation. We also state how graphs with properties similar to those exhibited by power grids have been modeled in the ERG literature.

A. General ERG model formulation

Let $\mathcal{G}_n$ be the collection of all undirected unweighted simple graphs on n nodes. We denote by $\mathbf{G}$ a random variable that takes values in $\mathcal{G}_n$. An *exponential random graph* (ERG) model [56] is a probability distribution on $\mathcal{G}_n$ of the form

$$\mathbb{P}_{\boldsymbol{\beta}}(\mathbf{G} = G) = \frac{\exp(\mathcal{H}_{\boldsymbol{\beta}}(G))}{Z_{\boldsymbol{\beta}}} = \frac{\exp\left(\sum_{i=1}^{r}\beta_i x_i(G)\right)}{Z_{\boldsymbol{\beta}}}, \tag{8}$$

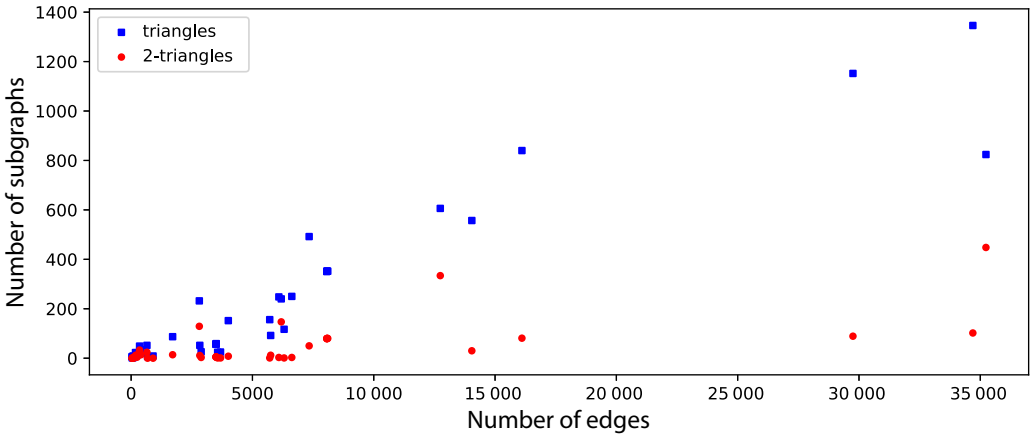

FIG. 3. The number t_1 of triangles (in blue) and the number t_2 of 2-triangles (in red) reported against the number of edges for various test networks (see Table V).

where $\boldsymbol{\beta} \in \mathbb{R}^r$ is a vector of parameters, $x_i : \mathcal{G}_n \mapsto \mathbb{R}$ are graph statistics of interest, also known as the *observables of the model*, and $Z_{\boldsymbol{\beta}} := \sum_{G \in \mathcal{G}_n} e^{\mathcal{H}_{\boldsymbol{\beta}}(G)}$ is a normalizing constant. The function $\mathcal{H}_{\boldsymbol{\beta}} : \mathcal{G}_n \to \mathbb{R}$ defined by $\mathcal{H}_{\boldsymbol{\beta}}(G) := \sum_{i=1}^{r} \beta_i x_i(G)$ is commonly called the *Hamiltonian* of the model. By definition, the density given in Eq. (8) is positive for all graphs $G \in \mathcal{G}_n$. Intuitively, graphs G with large values of $x_i(G)$ become less (more) likely if $\beta_i < 0$ ($\beta_i > 0$). By carefully choosing the parameters $\boldsymbol{\beta}$, the expected values $(\mathbb{E}[x_i(\mathbf{G})])_{i=1}^{r}$ of all the observables across the entire collection $\mathcal{G}_n$ can be tuned. In fact, the ERG density in Eq. (8) is the unique probability density $\mathbb{P}(G)$ over $\mathcal{G}_n$ that maximizes the Shannon entropy $S(\mathbb{P})$, defined as

$$S = -\sum_{G \in \mathcal{G}_n} \mathbb{P}(G) \ln \mathbb{P}(G), \tag{9}$$

subject to the constraints

$$\mathbb{E}[x_i(\mathbf{G})] = \bar{x}_i, \quad i = 1, \ldots, r, \tag{10}$$

where $\bar{x}_i$ is the desired average value of observable i. In other words, the distribution given in Eq. (8) does not contain more structured information than the constraints in Eq. (10).

Assuming that the target values $\bar{x}_i$, $i = 1, \ldots, r$, of the chosen r observables are given, one can tune the parameters $\boldsymbol{\beta}$ by solving the following system of equations, obtained by rewriting the constraints in Eq. (10) using Eq. (8):

$$\begin{aligned} \bar{x}_i = \mathbb{E}[x_i(\mathbf{G})] &= \frac{1}{Z_{\boldsymbol{\beta}}} \sum_{G \in \mathcal{G}_n} x_i(G) \mathrm{e}^{\mathcal{H}_{\boldsymbol{\beta}}(G)} \\ &= \frac{1}{Z_{\boldsymbol{\beta}}} \frac{\partial}{\partial \beta_i} \sum_{G \in \mathcal{G}_n} e^{\mathcal{H}_{\boldsymbol{\beta}}(G)} = \frac{1}{Z_{\boldsymbol{\beta}}} \frac{\partial Z_{\boldsymbol{\beta}}}{\partial \beta_i} \\ &= \frac{\partial F_{\boldsymbol{\beta}}}{\partial \beta_i}, \qquad i = 1, \ldots, r, \end{aligned} \tag{11}$$

where $F_{\boldsymbol{\beta}} := \ln Z_{\boldsymbol{\beta}}$ is the so-called *free energy* of the model. However, this strategy requires having a closed-form expression for the partition function $Z_{\boldsymbol{\beta}}$ or the free energy $F_{\boldsymbol{\beta}}$, which crucially depends on the choice for the Hamiltonian in Eq. (13) and is not available in most cases.

B. Proposed ERG models

In this paper, we propose using a synthetic grid procedure that samples random graphs using the probability distribution specified by an ERG model of the form given in Eq. (8). Our choice of observables is driven by the considerations made in Sec. II C and by a careful analysis of publicly available data from real grids collected and described in Ref. [50].

To fully introduce the model, we first need some preliminary definitions and additional notation. Consider a generic undirected graph $G = (V, E) \in \mathcal{G}_n$ with a fixed number n of nodes. Consistent with Sec. II A, we further assume that each node is (i) a generator node, (ii) a load node, or (iii) an interconnection node. Denote by $P, L, I \subset V$ the three corresponding subsets of generator, load, and interconnection nodes, respectively, so that $V = P \cup L \cup I$. Given two types of nodes $a, b \in \{P, L, I\}$, we denote by $E_{ab}(G) \subset E$ the subset of edges in the graph G that connect a node of type a to one of type b. Since we are working with an undirected graph, we have $E_{ab}(G) = E_{ba}(G)$. In this way, we obtain a partition of the edge set E as $E = E_{PP}(G) \cup E_{PL}(G) \cup E_{PI}(G) \cup E_{LL}(G) \cup E_{LI}(G) \cup E_{II}(G)$.

For each pair of node types $a, b \in \{P, L, I\}$, we denote the cardinality of the corresponding edge subset by $e_{ab}(G) := |E_{ab}(G)|$. Including these six edge counts as observables in the Hamiltonian allows us to simultaneously tune the average edge density and the average degree of the typical vertex of each bus type. However, as we will illustrate later in Sec. V C, the ERG model obtained by considering only these six edge counts as graph observables does not perform well, as the sampled graphs do

not have many of the desired properties mentioned in Sec. II C. In particular, this simple ERG model is unable to realistically replicate the clustering structure of power grids.

To overcome this limitation and capture the clustering properties of real power grids, we thus consider a more involved Hamiltonian that also includes the number of triangles t_1, as previously suggested in Ref. [57]. Recall from Sec. II A that the average clustering coefficient is defined precisely using the number of triangles. However, the resulting Hamiltonian has been proven to lead to degeneracy, i.e., the resulting ERG density assigns most of the probability mass either on (nearly) fully connected graphs or only on random bipartite graphs (see Ref. [58]), depending on the sign of the parameter associated with the triangle count.

Snijders *et al.* [59] have introduced a new class of models that exhibit the desired clustering coefficient and are not prone to degeneracy: the main idea is to include a more elaborate function of the number t_k of k-triangles with $k \geq 1$ rather than just the number t_1 of triangles. Specifically, we consider the so-called *alternating k-triangles statistic* of a graph G introduced in Ref. [59], which is defined as

$$u_\zeta(G) = 3t_1(G) - \frac{t_2(G)}{\zeta} + \frac{t_3(G)}{\zeta^2} - \ldots + (-1)^{n-3}\frac{t_{n-2}(G)}{\zeta^{n-3}}, \tag{12}$$

where ζ is a positive constant used to modulate the contribution of each k-triangle count to the observable. Intuitively, the alternating signs and decreasing weights in $u_\zeta(G)$ compensate each other, leading to highly clustered graphs that are not (nearly) fully connected or nearly empty.

To develop a new ERG model to generate synthetic power grids, some practical considerations are in order. For most of the power-grid topologies that are publicly available, the number of triangles t_1 increases linearly with the number of edges, while the number t_2 of 2-triangles grows at a significantly lower rate (see Fig. 3). More generally, in our analysis, we have found that the number of k-triangles is close to 0 for all power grids when $k > 2$. For this reason, rather than the involved alternating k-triangle statistic $u_\zeta(G)$, we have chosen to include in the ERG Hamiltonian two independent terms for the number of 1-triangles and 2-triangles (with two independent parameters β_{1t} and β_{2t}). The rationale behind this choice is the following: by ignoring all k-triangle counts for $k > 2$, we have enormously reduced the computational effort required by the model, while still maintaining the mitigating effect of the alternating signs of the parameters by imposing $\beta_{1t} \cdot \beta_{2t} \leq 0$, following [59]. The resulting Hamiltonian thus has $r = 8$ terms and reads

$$\begin{aligned}\mathcal{H}_{\boldsymbol{\beta}}(G) = {} & \beta_{PP}e_{PP}(G) + \beta_{PL}e_{PL}(G) + \beta_{PI}e_{PI}(G) \\ & + \beta_{LL}e_{LL}(G) + \beta_{LI}e_{LI}(G) \\ & + \beta_{II}e_{II}(G) + \beta_{1t}t_1(G) + \beta_{2t}t_2(G).\end{aligned} \tag{13}$$

This proposed ERG model assumes that the target values for these eight observables are known. To work with realistic target values, in the rest of the paper we adopt the following strategy: we consider a publicly available power grid G_0 with n nodes as a reference graph and use it to compute the target values as

$$\bar{x}_i = x_i(G_0), \quad i = 1, \ldots, r. \tag{14}$$

It is worth mentioning that, given the fact that the observables related to the edges in the Hamiltonian are negative, we expect the number of edges in the graphs generated with the model given in Eq. (13) to be on average directly proportional to the number of nodes in the graph, which is consistent with what has been observed for real power grids, as stated in Sec. II.

The addition of the 1- and 2-triangle terms in the Hamiltonian results in a partition function $Z_{\boldsymbol{\beta}}$ that has no closed-form expression, hence making it impossible to solve the systems of Eqs. (11) algebraically. Moreover, the collection $\mathcal{G}_n$ on which we have defined all ERG models so far contains many disconnected graphs, which are not of interest when modeling power-grid topologies, as we have argued in Sec. II C. However, it is not possible to explicitly take this requirement into account in the Hamiltonian, as there is no simple algebraic expression of the adjacency matrix that can capture the connectivity of the graph. Even if we were able to find a suitable proxy for connectivity to add as observable to the Hamiltonian, one should not forget that the ERG density imposes only *soft constraints* based on the observables. However, we want the connectedness of the graph to be a *hard constraint*, as the goal is to sample *connected* synthetic power grids. To accommodate this, we restrict the ERG model to the subsets of the *connected graphs* with n nodes, which we denote as $\mathcal{G}_{n,\text{conn}} \subset \mathcal{G}_n$. This is equivalent to sampling from the same ERG model in Eq. (13) but *conditional on* on the graph being connected.

The density of the ERG model obtained when restricting to a general subset $\mathcal{S} \subset \mathcal{G}_n$ is

$$\mathbb{P}_{\boldsymbol{\beta}_\mathcal{S}}(\mathbf{G} = G) = \frac{\exp(\mathcal{H}_{\boldsymbol{\beta}_\mathcal{S}}(G))}{Z_{\boldsymbol{\beta}_\mathcal{S}}} = \frac{\exp\left(\sum_{i=1}^r \beta_i x_i(G)\right)}{Z_{\boldsymbol{\beta}_\mathcal{S}}}, \tag{15}$$

for all $G \in \mathcal{S}$, where $Z_{\boldsymbol{\beta}_\mathcal{S}} := \sum_{G\in\mathcal{S}} e^{\mathcal{H}_{\boldsymbol{\beta}}(G)}$. We denote by $\boldsymbol{\beta}_\mathcal{S} = (\beta_1, \ldots, \beta_r)$ the new different set of parameters that the ERG model needs to express the target average for

each graph observable. In fact, when we restrict ourselves to a general subset $\mathcal{S} \subset \mathcal{G}_n$, the system of equations that determine the parameters given in Eq. (11) also changes. In particular, we have

$$\begin{aligned}\bar{x}_i = \mathbb{E}[x_i(\mathbf{G})] &= \frac{1}{Z_{\boldsymbol{\beta}_\mathcal{S}}} \sum_{G\in\mathcal{S}} x_i(G) \mathrm{e}^{\mathcal{H}_{\boldsymbol{\beta}}(G)} \\ &= \frac{1}{Z_{\boldsymbol{\beta}_\mathcal{S}}} \frac{\partial}{\partial \beta_i} \sum_{G\in\mathcal{S}} e^{\mathcal{H}_{\boldsymbol{\beta}}(G)} = \frac{1}{Z_{\boldsymbol{\beta}}} \frac{\partial Z_{\boldsymbol{\beta}_\mathcal{S}}}{\partial \beta_i} \\ &= \frac{\partial F_{\boldsymbol{\beta}_\mathcal{S}}}{\partial \beta_i}, i = 1, \ldots, r. \end{aligned} \tag{16}$$

In the special case with $\mathcal{S} = \mathcal{G}_{n,\mathrm{conn}}$ and the Hamiltonian as in Eq. (13), it is not possible to retrieve $Z_{\boldsymbol{\beta}_\mathcal{S}}$ analytically. Therefore, a numerical estimate is required to obtain a set of parameters that satisfy Eq. (16).

IV. PARAMETER ESTIMATION USING THE EQUILIBRIUM EXPECTATION ALGORITHM

In this section, we introduce a new method that we have used to estimate the parameters $\boldsymbol{\beta}$ that satisfy Eq. (16) for an ERG model formulation that includes a hard constraint as in Eq. (15). Before doing so, we briefly discuss the literature on ERG parameter estimation and relate it to our scheme.

A. Methods for ERG parameter estimation

Parameter estimation for ERG models in the presence of an intractable partition function is still an open problem [56]. The use of mean-field techniques gives unreliable results in related models such as spin glasses [60] and has been shown to work for ERG models only for specific values of the parameters that make them almost Erdős-Rényi models [58,61].

Many parameter-estimation approaches are based on MCMC methods. Such methods are widely used to sample from ERG models [62] and more generally from any probability distribution π. The Metropolis-Hastings (MH) algorithm [63] is an MCMC scheme that is particularly suitable when the target probability distribution π is known up to a constant factor, such as in the case of a Gibbs distribution or the ERG model given in Eq. (8).

The MH algorithm produces a sequence of samples from a Markov chain $\{X_t\}_{t\in\mathbb{N}}$ with specific transition probabilities that are obtained as the results of two steps, a proposal step and a subsequent acceptance step, each characterized by a different distribution. The proposal distribution $\mathbb{T}$ specifies the conditional probability $\mathbb{T}(s,s')$ of going from state s to state s'; the acceptance distribution $\mathbb{A}$ specifies the probability $\mathbb{A}(s,s')$ of accepting the proposed state s' if the chain previously resided in s. The transition probability can thus be rewritten as

$$\mathbb{P}_{\boldsymbol{\beta}}(s,s') = \mathbb{T}(s,s')\mathbb{A}(s,s'). \tag{17}$$

We will now specify the MH algorithm that we consider to estimate the parameters of the ERG model. Consider a general ERG model of the form given in Eq. (8), defined over a state space $\mathcal{S} \subseteq \mathcal{G}_n$. The target probability density π from which we want to sample is the ERG density $\mathbb{P}_{\boldsymbol{\beta}}$ given in Eq. (8).

In the classical MH algorithm for ERG models [62], only moves that prescribe the addition or removal of a single edge are allowed. In terms of the proposal distribution, this means that for every pair of graphs $G_t = (V, E_t), G_{t+1} = (V, E_{t+1}) \in \mathcal{S}$,

$$\mathbb{T}(G_t, G_{t+1}) > 0 \iff |E_t \bigtriangleup E_{t+1}| \leq 1. \tag{18}$$

We consider the following simple proposal distribution $\mathbb{T}(G_t, G_{t+1})$. We randomly choose a pair of nodes uniformly $(i,j) \in V \times V$ and if $(i,j) \notin E_t$, then we add the corresponding edge, obtaining a new set of edges $E_{t+1} = E_t \cup \{(i,j)\}$. Otherwise, we remove the selected edge and obtain a new graph with edge set $E_{t+1} = E_t \setminus \{(i,j)\}$. It is clear that this proposal distribution satisfies the condition given in Eq. (18).

Once a move is proposed, the acceptance probability is calculated using the desired target distribution given in Eq. (8) by

$$\mathbb{A}_{\boldsymbol{\beta}}(G_t, G_{t+1}) = \min\left\{1, \frac{\mathbb{P}_{\boldsymbol{\beta}}(G_{t+1})}{\mathbb{P}_{\boldsymbol{\beta}}(G_t)}\right\}, \tag{19}$$

with $\mathbb{P}_{\boldsymbol{\beta}}$ being the ERG probability distribution defined in Eq. (8). Note that computing the acceptance probability does not require knowledge of the partition function, since

$$\frac{\mathbb{P}_{\boldsymbol{\beta}}(G_{t+1})}{\mathbb{P}_{\boldsymbol{\beta}}(G_t)} = \frac{e^{\mathcal{H}_{\boldsymbol{\beta}}(G_{t+1})}}{e^{\mathcal{H}_{\boldsymbol{\beta}}(G_t)}}, \tag{20}$$

and, thus, it is possible to simplify the expression for the acceptance probability into

$$\mathbb{A}_{\boldsymbol{\beta}}(G_t, G_{t+1}) = \min\left\{1, e^{\mathcal{H}_{\boldsymbol{\beta}}(G_{t+1}) - \mathcal{H}_{\boldsymbol{\beta}}(G_t)}\right\}. \tag{21}$$

We denote by $\{X_t^{(\boldsymbol{\beta})}\}_{t\in\mathbb{N}}$ the Markov chain on the state space $\mathcal{S}$ defined by the MH algorithm. Note that the steady-state distribution π of $\{X_t^{(\boldsymbol{\beta})}\}_{t\in\mathbb{N}}$ exists and is unique since the chain is reversible. Furthermore, the use of Eq. (21) guarantees that π corresponds to the ERG density in Eq. (8).

A disadvantage of MCMC methods is that the chain must be close to stationarity to produce samples from the desired ERG distribution $\mathbb{P}_{\boldsymbol{\beta}}$. In practical terms, this means

that the chain needs to run for a sufficiently large number of steps, commonly referred to as the *mixing time* of the chain, so that the empirical distribution of the chain is close to its steady-state distribution.

One of the most widely used parameter-estimation approaches based on MCMC is the so-called *Markov-chain Monte Carlo maximum likelihood* introduced by Geyer [64]. The idea behind this procedure is to use samples from an MCMC as defined above, with an arbitrarily chosen set of parameters $\boldsymbol{\beta}_0$ to approximate the log-likelihood of the distribution $\mathcal{L}_{\boldsymbol{\beta}}$ with an empirical one $\mathcal{L}_{N,\boldsymbol{\beta}_0}$, where N is the number of samples drawn from the steady-state distribution of the chain, to retrieve the set of parameters that satisfy Eq. (11). This method theoretically guarantees the asymptotic convergence of $\mathcal{L}_{N,\boldsymbol{\beta}_0}$ to $\mathcal{L}_{\boldsymbol{\beta}}$ when the number of samples N approaches infinity. However, the convergence is slow if $\boldsymbol{\beta}_0$ is too far from the target one. If this happens, the literature suggests reiterating the procedure many times, using the end point of the previous iteration as the starting parameter for the next iteration. Although coming with asymptotic convergence guarantees, the maximum-likelihood estimation through MCMC can be computationally unfeasible in our context, especially when the number of nodes is too large. At each iteration of the method with different parameters, a large number of samples must be taken from the steady-state distribution of the chain after it has reached the mixing time. This could be impractical for large n. For ERG models for dense graphs, the mixing time has been shown to be of the order $\mathcal{O}(n^2 \log n)$ [65]; however, currently there are no similar results for ERG models on sparse graphs, i.e., ERG densities the ensemble of which has an average number of edges $\langle m \rangle$ of the order $\mathcal{O}(n)$.

From a Bayesian perspective, there are many possible methods that have been proposed and used in the literature to tackle this issue (see, e.g., Ref. [66]) but they are known to scale poorly as the number of nodes increases. Nevertheless, the MCMC maximum-likelihood method is widely used in the ERG community, being also the main method used in popular estimation libraries for ERGMs such as the ERGM package [67,68] for the **R** software [69]. This package estimates the parameters of the models using maximum-likelihood estimation, which is approximated using either the MCMC MLE, which we described before, or the maximum-pseudolikelihood estimation (often referred to as MPLE, first introduced in Ref. [70]), or combination of the two. However, using any of these approaches while working in the space of connected graphs $\mathcal{G}_{n,\text{conn}}$ is highly nontrivial. Moreover, enforcing sparsity in the graphs while also imposing the hard constraint of connectivity could be computationally unfeasible for the aforementioned methods, motivating the need for new procedures for this specific problem.

ALGORITHM 1. Equilibrium Expectation for a constrained chain.

Input: the starting graph $G_0 \in \mathcal{S}$; θ; $\boldsymbol{\beta}_0$; T; α; c
Set $G_0 := G_0$ and $t = 0$
for $t = 1, \ldots, T$ **do**
  Sample a random pair $i, j, i \neq j$ uniformly at random from $1, \ldots, n$
  if $(i, j) \in E_{t-1}$ **then**
    Remove the edge (i, j) and consider a new edge set $\tilde{E}_t = E_{t-1} \setminus (i, j)$
    if $\tilde{G}_t = (V, \tilde{E}_t) \in \mathcal{S}$ **then**
      Accept $\tilde{G}_t$ with MH acceptance rate, *i.e.*, $G_t = \tilde{G}_t$ w.p. $\min\left\{1, \mathbb{P}_{\boldsymbol{\beta}}(G_t)/\mathbb{P}_{\boldsymbol{\beta}}(G_{t-1})\right\}$
    else
      Reject G_t *i.e.*, $G_t = G_{t-1}$
    end if
  else
    Add the edge (i, j) and consider a new edge set $\tilde{E}_t = E_{t-1} \cup (i, j)$
    Accept $\tilde{G}_t$ with MH acceptance rate, *i.e.*, $G_t = \tilde{G}_t$ w.p. $\min\left\{1, \mathbb{P}_{\boldsymbol{\beta}}(G_t)/\mathbb{P}_{\boldsymbol{\beta}}(G_{t-1})\right\}$
  end if
  Update $\boldsymbol{\beta}_t$ according to rule (22)
end for
Compute the estimate $\bar{\boldsymbol{\beta}} := \bar{\boldsymbol{\beta}}(T) = \frac{1}{T}\sum_{t=1}^{T} \boldsymbol{\beta}_t$
Output: $\bar{\boldsymbol{\beta}}$

B. Estimation algorithm for constrained chains

In view of the above considerations, we have decided to resort to a variation of the *equilibrium-expectation* (EE) algorithm [71] to estimate the parameters of our ERG model. The EE algorithm uses a modified MH MCMC model, as defined in Eq. (17), in which the parameters $\boldsymbol{\beta}$ of the ERG model are dynamically adjusted. In contrast to other MCMC methods, it is based on the properties of the chain at equilibrium rather than on drawing a large number of samples from the chain itself, thus decreasing the computational burden. The pseudocode for the designed algorithm is given in Algorithm 1.

The initial state $X_0 = G_0$ of the Markov chain associated with the EE algorithm must be drawn from the desired steady-state distribution to exploit the properties of the chain at equilibrium. In our setting, this requirement is satisfied by taking as an initial state the graph corresponding to the power grid used as a reference for the Hamiltonian given in Eq. (13).

At step t of the chain, the parameters $\boldsymbol{\beta}$ are updated according to some update rule if $t \equiv 0 \pmod{\theta}$, with θ being a user-defined variable that controls how often the update occurs. The update rule proposed in Ref. [71] works as follows. Let $\boldsymbol{\beta}_t = (\beta_1^t, \ldots, \beta_r^t)$ be the parameters associated with the observables $x_1(G), x_2(G), \ldots, x_r(G)$ at time t.

At step $t+1$, the parameters are updated as

$$\beta_i^{t+1} = \beta_i^t + \alpha \cdot \max(|\beta_i^t|, c) \cdot \text{sign}[x_i(G_0) - x_i(G_t)] \quad (22)$$

if $t \equiv 0 \pmod{\theta}$ and $\beta_i^{t+1} = \beta_i^t$ otherwise. The parameter $\alpha > 0$ is the learning rate and $c > 0$ is a control parameter that ensures that the algorithm does not get stuck when $|\beta_i^t| \ll 1$.

This approach alone does not allow for sampling from a constrained space such as $\mathcal{G}_{n,\text{conn}}$. To overcome this, we combine the EE algorithm with the modified MH-type algorithm proposed by Grey *et al.* in Ref. [72], designed specifically to sample from constrained sets of graphs.

Consider a nonempty connected subset $\mathcal{S} \subset \mathcal{G}_n$ and a Hamiltonian $H_{\boldsymbol{\beta}}$ with r observables. We will prove that if the Markov chain $\{X_t\}_{t\in\mathbb{N}}$ on $\mathcal{S}$ defined by the EE algorithm (i) satisfies Eq. (18) and

$$\mathbb{T}(G, G') = 0 \quad \forall\, G \in \mathcal{S},\ \forall\, G' \notin \mathcal{S}, \quad (23)$$

and (ii) uses the acceptance rule as in Eq. (21), then as $t \to \infty$, the parameter process $\boldsymbol{\beta}^t$ converges to the solution $\boldsymbol{\beta}_{\mathcal{S}}$ of the equations

$$\bar{x}_i(G_0) = \frac{1}{Z} \sum_{G\in\mathcal{S}} x_i(G) \mathrm{e}^{\mathcal{H}_{\boldsymbol{\beta}_{\mathcal{S}}}(G)}, \quad i \in \{1, \ldots, r\}. \quad (24)$$

In our case, we take $\mathcal{S}$ to be the set of connected graphs of size n, i.e., $\mathcal{S} = \mathcal{G}_{n,\text{conn}}$. For Eq. (23) to hold, the Markov chain should be defined to have nonzero transition probabilities only between connected graphs. To this end, we choose the following proposal density $\mathbb{T}(G_t, G_{t+1})$: either an edge chosen uniformly from the set of missing edges $E \setminus E_t$ is added or an edge chosen uniformly among the edges in E_t that can be removed without disconnecting the graph is removed.

This algorithm takes as input a starting graph G_0 for the chain, the number of steps θ of the Markov chain after which the parameters are updated, the starting parameters $\boldsymbol{\beta}_0$, the hyperparameters α and c used for the update rule given in Eq. (22), and the maximum number of iterations T. The EE algorithm works as follows: each proposed move is either the addition of an edge or the removal of an existing edge that does not disconnect the graph. In either case, the move is accepted according to the standard MH acceptance rule.

After every θ transitions of $\{X_t\}_{t\in\mathbb{N}}$, each parameter β_i is updated simultaneously using the update rule given in Eq. (22). The limit of the sequence $\boldsymbol{\beta}_t$ generated by the EE algorithm does not depend on the initial values $\boldsymbol{\beta}_0$. In fact, later, in Theorem 1, we will show that the limit is unique. However, the choice of $\boldsymbol{\beta}_0$ in general affects the convergence speed of the algorithm but we have not investigated this numerically. The starting values of $\boldsymbol{\beta}_0$ could be obtained by using, e.g., the contrastive-divergence method, as suggested in Ref. [71].

The following theorem summarizes the results concerning the convergence of the method.

Theorem 1. Consider the coupled stochastic processes $(X_t, \boldsymbol{\beta}_t)_{t\in\mathbb{N}}$ returned by Algorithm 1. Let $\{X_t\}_{t\in\mathbb{N}}$ be a Markov chain with transition probabilities defined as in Eq. (17) on a nonempty and connected subset of graphs $\mathcal{S} \subseteq \mathcal{G}_n$, with a proposal $\mathbb{T}$ as in Eq. (18) and acceptance probability $\mathbb{A}_{\boldsymbol{\beta}_t}$ as in Eq. (21) and let $\{\boldsymbol{\beta}_t\}_{t\geq 0}$ be the r-dimensional stochastic process describing the evolution of the ERG parameters over time using the update rule given in Eq. (22). If the starting point X_0 is drawn from the steady-state distribution of $\{X_t\}_{t\in\mathbb{N}}$ and the learning rate α in the update rule is small enough, then for any $\boldsymbol{\beta}_0$,

$$\lim_{T\to\infty} \bar{\boldsymbol{\beta}}(T) = \lim_{T\to\infty} \frac{1}{T} \sum_{t=1}^{T} \boldsymbol{\beta}_t = \boldsymbol{\beta}_{\mathcal{S}}, \quad (25)$$

where $\boldsymbol{\beta}_{\mathcal{S}}$ is the set of parameters satisfying Eq. (16).

Proof. In this proof, we will write $\mathbb{P}_{\boldsymbol{\beta}}$ and $\mathbb{A}_{\boldsymbol{\beta}}$ instead of $\mathbb{P}_{\boldsymbol{\beta}_t}$ and $\mathbb{A}_{\boldsymbol{\beta}_t}$, making the time dependency of the parameters $\boldsymbol{\beta}$ implicit to keep the notation light. Similarly, we will write $\chi_i(t+1, t)$ instead of $(x_i(G_{t+1}) - x_i(G_t))$

First, we prove that the chain defined by the algorithm in the restricted state space $\mathcal{S}$ is irreducible and aperiodic. The key observation is that starting from any graph, it is possible to reach the complete graph K_n in a finite number of moves by adding all the missing edges one by one. These trajectories are possible in the subspace $\mathcal{S}$ since it is closed with respect to edge additions. Furthermore, since every edge addition can be "reversed," the corresponding symmetric edge removal is allowed in $\mathcal{S}$. For any pair of connected graphs $G, G' \in \mathcal{S}$, we can build a trajectory in $\mathcal{S}$ between them by first connecting G to K_n using the trajectory described above and then using the reverse trajectory from K_n to G'. This property is independent of the choice of parameters $\boldsymbol{\beta}_t$; therefore, the considered Markov chain $\{X_t\}_{t\in\mathbb{N}}$ is thus irreducible. The aperiodicity readily follows from the way in which we have defined $\mathbb{A}_{\boldsymbol{\beta}}$, which assigns a strictly positive probability to stay in one configuration for more than one step.

Our algorithm fits within the framework of Ceperley and Dewing [73], which we now briefly discuss. The goal of Ref. [73] is to determine an explicit expression for the acceptance probability in MCMC in such a way that the algorithm still samples from the desired (stationary) distribution π in the setting where the energy difference Δ between the current state and the next proposed state is perturbed by noise. This also fits our setting; we have $\Delta = \mathcal{H}_{\boldsymbol{\beta}_{t+1}}(G_{t+1}) - \mathcal{H}_{\boldsymbol{\beta}_t}(G_t)$ and for each parameter β_j we have that (β_j^t) for $t = 0, 1, 2, \ldots, T$ is a sequence of random

variables. Note that each β_j^t is adapted to the filtration generated by the initial condition and the acceptance choices up to time t. Consequently, the energy difference between graphs G_t and G_{t+1},

$$\Delta = \Delta(G_t, G_{t+1}) := \sum_{j=1}^{r} \beta_j^t \chi_j(t+1, t), \tag{26}$$

is also random. More specifically, as stated in Ref. [73], a central-limit theorem argument suggests that for large enough T, for each j, the distribution of β_j^t is approximately normally distributed with mean β_j and variance α^2. This is indeed confirmed by the numerical experiments with the update rule given in Eq. (22) conducted in Ref. [74]. Assuming that $(\beta_j^t)_{j=1}^r$ are jointly normally distributed (with a nontrivial covariance matrix), we have

$$\Delta \sim \mathcal{N}(\mu_\Delta, \sigma_\Delta^2), \tag{27}$$

where $\mu_\Delta := \sum_{j=1}^r \beta_j \chi_j(t+1, t)$. The variance of Δ is

$$\sigma_\Delta^2 := \mathrm{Var}\Big(\sum_{j=1}^{r} \beta_j^t \chi_j(t+1,t)\Big) = \sum_{j=1}^{r} \mathrm{Var}\Big(\beta_j^t \chi_j(t+1,t)\Big) + \sum_{i\neq j} \mathrm{Cov}\Big(\beta_i^t \chi_i(t+1,t), \beta_j^t \chi_j(t+1,t)\Big). \tag{28}$$

We now aim to find an upper bound on the variance. Consider the case in which the sum of the covariance terms in Eq. (28) is greater than 0. Applying the Cauchy-Schwarz inequality, we have the following result:

$$\sum_{i\neq j} \mathrm{Cov}\Big(\beta_i^t \chi_i(t+1,t), \beta_j^t \chi_j(t+1,t)\Big) \leq \sum_{i\neq j} \sqrt{\mathrm{Var}\Big(\beta_i^t \chi_i(t+1,t)\Big) \cdot \mathrm{Var}\Big(\beta_j^t \chi_j(t+1,t)\Big)}. \tag{29}$$

By plugging Eq. (29) into Eq. (28), we obtain an upper bound that depends only on the variances of the single differences in the Hamiltonian given in Eq. (30):

$$\mathrm{Var}\Big(\sum_{j=1}^{r} \beta_j^t \chi_j(t+1,t)\Big) \leq \sum_{j=1}^{r} \mathrm{Var}\Big(\beta_j^t \chi_j(t+1,t)\Big) + \sum_{i\neq j} \sqrt{\mathrm{Var}\Big(\beta_i^t \chi_i(t+1,t)\Big)\mathrm{Var}\Big(\beta_j^t \chi_j(t+1,t)\Big)}. \tag{30}$$

Now note that the terms $\chi_j(t+1,t)$ are bounded for each j and for every t, since they represent time differences in graph statistics the maximum of which can be easily computed given the j th statistic formula. In fact, we can bound the differences in the observables by writing

$$|x_j(G_{t+1}) - x_j(G_t)| \leq Q_{\max}, \quad \forall j, \forall t, \tag{31}$$

where $Q_{\max}$ is the maximum absolute difference of any observable between two graphs differing by one edge. In our formulation, with the Hamiltonian defined in Eq. (13), we expect $Q_{\max}$ to be equal to the maximum number of 2-triangles that can be created or removed by the addition or removal of a single edge, which is of the order $\mathcal{O}(k_{\max}^2)$, where $k_{\max}$ is the maximum node degree of G_t. Since we are in a sparse regime, we expect $k_{\max}$ to be of order $o(n)$ (as stated in Sec. II C, for power grids it has been observed that $2 < \langle k \rangle < 3$).

Thus, we can substitute their contribution in the variance with the constant $Q_{\max}$, obtaining

$$\mathrm{Var}\Big(\sum_{j=1}^{r} \beta_j^t \chi_j(t+1,t)\Big) \leq \sum_{j=1}^{r} Q_{\max}^2 \mathrm{Var}\Big(\beta_j^t\Big) + \sum_{i\neq j} Q_{\max}^2 \sqrt{\mathrm{Var}\Big(\beta_i^t\Big) \cdot \mathrm{Var}\Big(\beta_j^t\Big)}. \tag{32}$$

Now note that the upper bound for the variance depends solely on the variances of the parameters. With the update rule described in Eq. (22) and Algorithm 1, the variances of the parameters, as stated in Ref. [74], can be assumed to be approximately equal to the square of the learning rate α, leading to the following upper bound:

$$\mathrm{Var}\Big(\sum_{j=1}^{r} \beta_j^t \chi_j(t+1,t)\Big) \leq \sum_{j=1}^{r} Q_{\max}\alpha^2 + \sum_{i\neq j} Q_{\max}\alpha^2. \tag{33}$$

In Ref. [73], the authors determine an acceptance probability that satisfies the *average* detailed balance equation of the chain. In our setting, the average detailed balance equation is

$$\pi(G_t)\mathbb{A}_{\boldsymbol{\beta}}(G_t, G_{t+1}) = \pi(G_{t+1})\mathbb{A}_{\boldsymbol{\beta}}(G_{t+1}, G_t), \tag{34}$$

where G_t and G_{t+1} differ by one edge. When G_t and G_{t+1} differ by more than one edge, the average detailed balance equations are always trivially satisfied. In Eq. (34), $\mathbb{A}_{\boldsymbol{\beta}}(G_t \to G_{t+1})$ is the *average* acceptance probability of the transition from G_t to G_{t+1}, where the expectation is taken with respect to the law of $(\beta_i^t)_{i=1}^r$. The results in Ref. [73] imply that, when $\Delta(G_t, G_{t+1})$ follows a normal distribution with *known* variance σ_Δ^2, the acceptance

probability

$$\mathbb{A}_{\boldsymbol{\beta}}(G_t, G_{t+1}; \sigma_\Delta) = \exp\left(-\Delta(G_t, G_{t+1}) - \sigma_\Delta^2/2\right) \wedge 1 \tag{35}$$

solves Eq. (34).

The issue with this is that, in general, σ_Δ in Eq. (35) cannot be computed explicitly. However, given the upper bound for the variance described in Eq. (33), if one chooses a small enough learning rate α (possibly at the cost of a larger t), the term σ_Δ^2 can be neglected, leading to the usual acceptance probability,

$$\mathbb{A}_{\boldsymbol{\beta}}(G_t, G_{t+1}) = \exp\left(-\Delta(G_t, G_{t+1})\right) \wedge 1. \tag{36}$$

With the acceptance probability defined by Eq. (36) and the constraint that every state of the chain trajectory should lie in the subspace $\mathcal{S}$ described before, we can follow the same steps as in Ref. [72]. First, since each state of the trajectory is required to lie in $\mathcal{S}$, we have the following:

$$\frac{\mathbb{P}_{\boldsymbol{\beta}}(G_{t+1})}{\mathbb{P}_{\boldsymbol{\beta}}(G_t)} = \frac{\mathbb{P}_{\boldsymbol{\beta}}(\mathbf{G} = G_{t+1} | \mathbf{G} \in \mathcal{S})}{\mathbb{P}_{\boldsymbol{\beta}}(\mathbf{G} = G_t | \mathbf{G} \in \mathcal{S})}, \tag{37}$$

which can be rewritten as

$$\begin{aligned} &\frac{\mathbb{P}_{\boldsymbol{\beta}}(\mathbf{G} = G_{t+1}, \mathbf{G} \in \mathcal{S})}{\mathbb{P}_{\boldsymbol{\beta}}(\mathbf{G} = G_t, \mathbf{G} \in \mathcal{S})} \cdot \frac{\mathbb{P}_{\boldsymbol{\beta}}(\mathbf{G} \in \mathcal{S})}{\mathbb{P}_{\boldsymbol{\beta}}(\mathbf{G} \in \mathcal{S})} \\ &\quad = \frac{\mathbb{P}_{\boldsymbol{\beta}}(\mathbf{G} = G_{t+1}, \mathbf{G} \in \mathcal{S})}{\mathbb{P}_{\boldsymbol{\beta}}(\mathbf{G} = G_t, \mathbf{G} \in \mathcal{S})}. \end{aligned} \tag{38}$$

By definition of the moves in Algorithm 1, we have $\mathbb{P}_{\boldsymbol{\beta}}(\mathbf{G} = G, \mathbf{G} \in \mathcal{S}) = \mathbb{P}_{\boldsymbol{\beta}}(\mathbf{G} = G)$ because the trajectory is always restricted to lie inside $\mathcal{S}$. Therefore, we rewrite Eq. (38) as follows:

$$\frac{\mathbb{P}_{\boldsymbol{\beta}}(G_{t+1})}{\mathbb{P}_{\boldsymbol{\beta}}(G_t)} = \frac{\mathbb{P}_{\boldsymbol{\beta}}(\mathbf{G} = G_{t+1})}{\mathbb{P}_{\boldsymbol{\beta}}(\mathbf{G} = G_t)}, \tag{39}$$

which in turn can be plugged into Eq. (36), leading to the following acceptance probability:

$$\mathbb{A}_{\boldsymbol{\beta}}(G_t, G_{t+1}) = \min\left\{1, \frac{\mathbb{P}_{\boldsymbol{\beta}}(\mathbf{G} = G_{t+1})}{\mathbb{P}_{\boldsymbol{\beta}}(\mathbf{G} = G_t)}\right\}, \tag{40}$$

which is the same acceptance probability as for the MH algorithm without constraint. By construction, this acceptance probability guarantees that $\pi(G) = \mathbb{P}_{\boldsymbol{\beta}_\mathcal{S}}(\mathbf{G} = G | \mathbf{G} \in \mathcal{S})$, which coincides with the desired probability distribution given in Eq. (15). Hence, since the chain is irreducible and aperiodic for $t \to \infty$, Algorithm 1 converges to the distribution of interest given in Eq. (15) if the learning rate α is close to 0 with the update rule as in Eq. (22) and with the acceptance probability given in Eq. (40). ■

ALGORITHM 2. Connected graph generation [72].

Input: A connected graph $G_0 = (V, E_0)$, set of parameters $\boldsymbol{\beta}$, maximum number of iterations T
for $t = 1, \ldots, T$ **do**
  Generate a random pair (i, j), $i \neq j$, $i, j \in V$
  if $(i, j) \in E$ **then**
    Remove the edge (i, j) and consider $\tilde{E}_t = E_{t-1} \setminus \{(i, j)\}$, obtaining a new graph $\tilde{G}_t := (V, \tilde{E}_t)$
    if $\tilde{G}_t$ is connected **then**
      Accept $\tilde{G}_t$ with MH acceptance rate, *i.e.*, $G_t = \tilde{G}_t$ w.p. $\min\left\{1, \mathbb{P}_{\boldsymbol{\beta}}(G_t)/\mathbb{P}_{\boldsymbol{\beta}}(G_{t-1})\right\}$, and save it in graph output list G
    else
      Reject G_t *i.e.*, $G_t = G_{t-1}$
    end if
  else
    Add the edge (i, j) and consider $\tilde{E}_t = E_{t-1} \cup \{(i, j)\}$, obtaining a new graph $\tilde{G}_t := (V, \tilde{E}_t)$
    Accept $\tilde{G}_t$ with MH acceptance rate, *i.e.*, $G_t = \tilde{G}_t$ w.p. $\min\left\{1, \mathbb{P}_{\boldsymbol{\beta}}(G_t)/\mathbb{P}_{\boldsymbol{\beta}}(G_{t-1})\right\}$, and save it in graph output list G
  end if
end for
Output: Graph list G

After having estimated the set of parameters $\bar{\boldsymbol{\beta}}$ using Algorithm 1, we can then use Ref. [72, Algorithm 2] to obtain an ensemble of graphs sampled from the probability distribution $\mathbb{P}_{\boldsymbol{\beta}_\mathcal{S}}$ as in Eq. (15). This algorithm is a modified MH algorithm that generates only connected graphs; for completeness, we report the pseudocode in Algorithm 2.

V. NUMERICAL RESULTS

In this section, we give details on the tuning of the proposed ERG model and present some numerical results. More specifically, in Sec. V A, we detail our implementation of the proposed variant of the EE algorithm, while in Sec. V B, we report several statistics of the synthetic grids obtained with the proposed ERG-based procedure against the reference topology. Lastly, in Sec. V C, we compare the performance of our proposed ERG model with the simpler ERG model the Hamiltonian of which does not have the 1- and 2-triangle terms.

Our implementation and all our experiments are available at [75]. We believe that there is room for significant computational speed-ups in our implementation but these are beyond the scope of our paper and are left for future work. The experiments in this section have been performed on a laptop with an Intel® Core™ i7 CPU and 16 GB of RAM.

TABLE I. The hyperparameters used for the EE algorithm when using the 300 IEEE network as reference grid.

T	α	c	θ
20 000 000	0.001	0.001	100

A. ERG parameter tuning

In this section, as an illustration, we briefly show how we have tuned the parameters for an ERG model that aims to generate synthetic power grids with topological properties similar to a grid given in input. The grid taken as reference is the 300 IEEE network, a medium-sized benchmark grid with 300 nodes and 409 edges (see Table V).

In Table I, we report the hyperparameters of Algorithm 1 used for the estimation of the parameters of the ERG model with Hamiltonian as in Eq. (13) and the topology of the 300 IEEE network as reference for the values of the observables.

Since we have no knowledge of the distribution given in Eq. (15) before parameter estimation, in all our experiments we use as a starting point for the estimation in Algorithm 1 the topology of the reference grid G_0. Furthermore, the statement of Theorem 1 holds asymptotically as $T \to \infty$ but, in practice, we need to resort to an approximation of Eq. (25). If T is large enough, we can estimate the target parameters as in Ref. [74], using

$$\bar{\boldsymbol{\beta}} = \frac{1}{T - t_B} \sum_{t=t_{B+1}}^{T} \boldsymbol{\beta}_t, \tag{41}$$

where t_B is the so-called *burn-in time*, i.e., a time after which we say that the chain $\{\boldsymbol{\beta}(t)\}_{t\in\mathbb{N}}$ is roughly independent of the initial condition $\boldsymbol{\beta}_0$.

For simplicity purposes, in all of our experiments, we pick the burn-in time in a heuristic way by setting $t_B = 0.75 \cdot T$, effectively keeping only the last quarter of the trajectories to estimate $\bar{\boldsymbol{\beta}}$.

In Fig. 4, we show the trajectories of the various components of the vector $\boldsymbol{\beta}_t$ obtained using Algorithm 1. For illustration purposes, we display the values only after each nontrivial update [i.e., $\boldsymbol{\beta}_{k\cdot\theta}, k \in \mathbb{N}$, cf. Eq. (22)], effectively displaying only the nonpiecewise constant parts of the trajectories. For this experiment, the percentage of accepted moves is approximately 1.5%, and the percentage of rejected proposals due to the connectivity constraint is approximately 0.3%.

Looking at Fig. 4, the burn-in time can be graphically interpreted as the number of update iterations $\bar{t}$ after which each parameter trajectory has passed the "elbow point" and starts to oscillate around a constant value.

B. Topological properties of generated synthetic grids

We present here the results of our proposed sampling method on a few benchmark grids of different sizes. For each grid, we sample new graph topologies using the Markov chain described in Algorithm 2, with grid-specific parameters $\bar{\boldsymbol{\beta}}$ tuned with Algorithm 1.

To reduce undesirable correlations among the sampled graphs, we have applied a standard chain-thinning criterion to the list of graphs produced by Algorithm 2. More specifically, we progressively scan this list of samples and retain only the graphs the adjacency matrix of which is at a Hamming distance of at least $2n$ from the adjacency matrix of the last retained graph.

In Table II, we compare the sampled graphs with the initial reference power grids G_0, with respect to various topological properties described in Sec. II. We report the average of each quantity over the collection of sampled graphs, as well as the corresponding standard deviation. The whole procedure, complete with parameter estimation and grid ensemble generation, took 25 min for the 118 IEEE network, 70 min for the 300 IEEE network, and 250 min for the 1354 PEGASE network.

All generated synthetic topologies are connected, have a realistic bus-type assignment as defined in Refs. [10,37], and exhibit average values close to those of the grid of reference for all observables included in the Hamiltonian. We observe that for the topological properties that were not explicitly included in our Hamiltonian, e.g., the average path length and the algebraic connectivity, the obtained average values, although not exactly matching the real ones, are within the range of values considered as realistic in previous literature (see Ref. [11,38]) and are comparable to those observed in the real grids of similar size reported in Appendix A. A direct comparison with other models in the literature is difficult due to the different grids used as reference and the different scope (comparing a spatially embedded model with a purely topological one would be uninformative at best, due to the differences in the problems considered). However, we emphasize that our results for the topological properties of interest are consistent with those obtained by topological methods using reference grids of similar size to the one we have considered [5,31,38].

As mentioned above, our approach only considers the bus-type assignment and does not take into account any other electrical properties. This is because we believe that attempting to include such properties in an ERG formulation would be highly challenging and inefficient. To effectively include these properties in the generated grids, one could rely on the methods proposed in the literature that can be used to incorporate electrical parameters into an existing topology in a realistic way [76,77]. These approaches typically involve sampling operational data

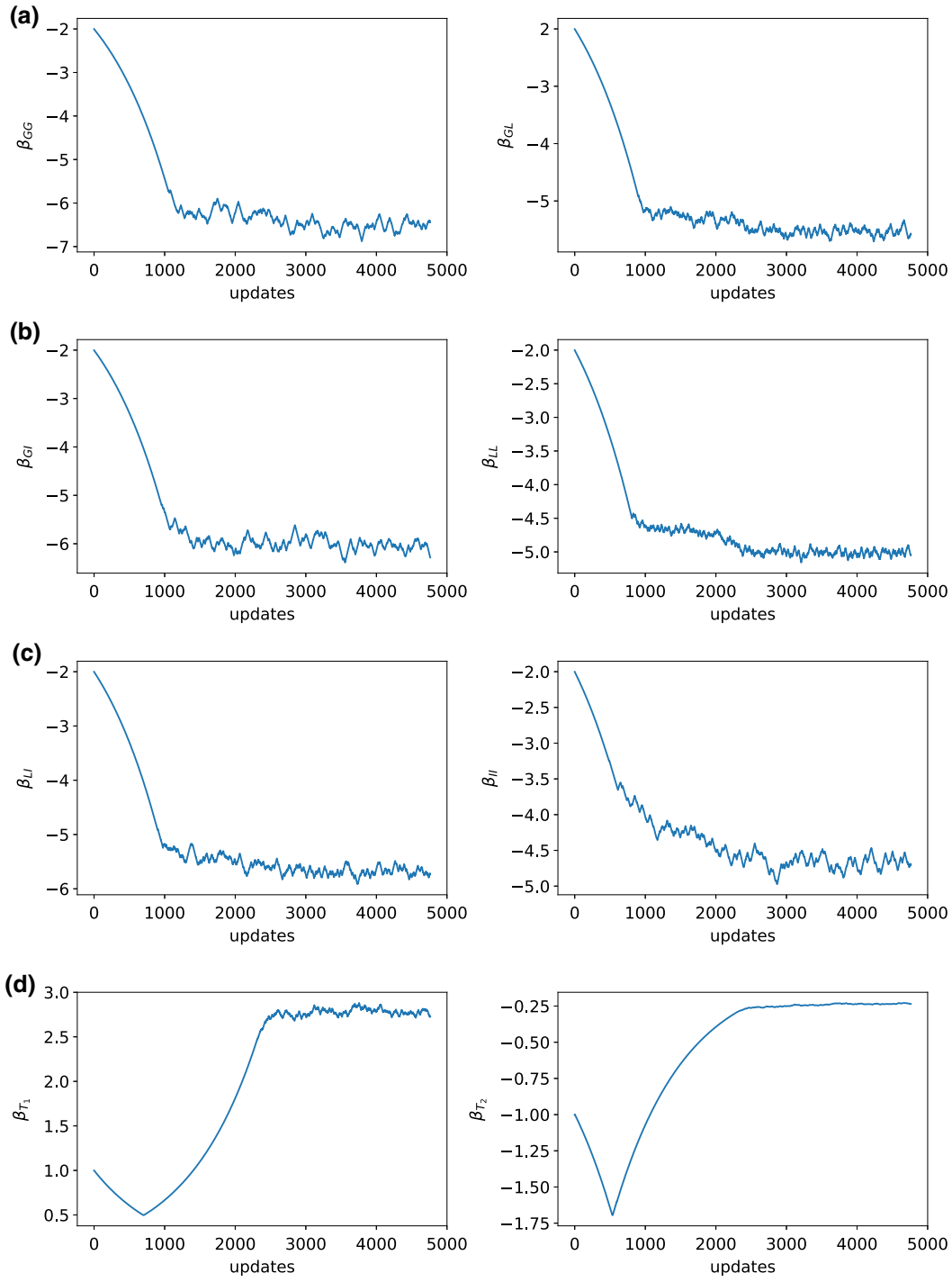

FIG. 4. The trajectories of the $\boldsymbol{\beta}_{k\cdot\theta}, k \in \mathbb{N}$ process describing the ERG model parameter updates over time using the EE algorithm when using the `300 IEEE` network as the reference grid. (a) The evolution over time of the parameters β_{PP} (left) and β_{PL} (right). (b) The evolution over time of the parameters β_{PI} (left) and β_{LL} (right). (c) The evolution over time of the parameters β_{LI} (left) and β_{II} (right). (d) The evolution over time of the parameters β_{1t} for the number of triangles (left) and β_{2t} for the number of 2-triangles (right).

from probability distributions estimated from real grids and attaching the sampled values to the topology, while ensuring that realism constraints are satisfied. The same methods could be applied to the topologies generated by our model.

C. Ablation study

To show the crucial role that including the number of 1- and 2-triangles as observables in the Hamiltonian plays, we present here a simpler ERG model that only accounts for the edge count and bus-type assignment. Specifically,

TABLE II. A comparison of the synthetic grid statistics obtained from the ERG model with the corresponding actual values in the reference grid G_0. The reported graph observables are as follows: m, average number of edges; $\langle k_P \rangle$, average degree of generators; $\langle k_L \rangle$, average degree of loads; $\langle k_I \rangle$, average degree of interconnections; APL, average shortest path length; λ_2, algebraic connectivity; C, clustering coefficient. The averages and deviations have been computed on an ensemble composed of 1100 samples, obtained after chain thinning.

Power grid G_0	Metric type	m	$\langle k_P \rangle$	$\langle k_L \rangle$	$\langle k_I \rangle$	APL	λ_2	C
`118 IEEE`	Actual	179	3.56	2.5	3.25	6.3	0.027	0.165
	Sample average	180	3.61	2.52	3.33	4.67	0.10	0.11
	Sample standard	(6.89)	(0.36)	(0.20)	(0.70)	(0.42)	(0.04)	(0.03)
`300 IEEE`	Actual	409	1.95	3.00	2.15	9.93	0.009	0.085
	Sample average	413	1.93	3.06	2.16	6.44	0.049	0.076
	Sample standard	(10)	(0.13)	(0.17)	(0.24)	(0.39)	(0.019)	(0.015)
`1354 PEGASE`	Actual	1710	2.58	1.08	2.53	11.15	0.005	0.056
	Sample average	1723	2.53	1.16	2.51	9.56	0.022	0.051
	Sample standard	(29)	(0.09)	(0.17)	(0.04)	(0.27)	(0.007)	(0.006)

we consider the following Hamiltonian:

$$\mathcal{H}_{\boldsymbol{\beta}}(G) = \beta_{PP}|E_{PP}| + \beta_{PL}|E_{PL}| + \beta_{PI}|E_{PI}| + \beta_{LL}|E_{LL}| + \beta_{LI}|E_{LI}| + \beta_{II}|E_{II}|, \tag{42}$$

which is the same formulation as in Eq. (13) but without the counts of the 1- and 2-triangles as observables. This simpler Hamiltonian allows us to calculate the partition function $Z_{\boldsymbol{\beta}}$ explicitly and thus derive in a close form the optimal parameters given the target observables. In fact, the partition function of this simpler model is

$$Z_{\boldsymbol{\beta}} = \prod_{i \in I} (1 + e^{\beta_i})^{M(E_i)}, \tag{43}$$

where $I = \{PP, PL, LI, LL, LI, II\}$ and $M(E_i)$ is a function that computes the maximum possible number of edges of type i. The theoretical details of the derivations of Eq. (43) are presented in Appendix B. In other words, if $E_i = E_{ab}$ is the subset of edges that connect nodes of type a with nodes of type b, then

$$M(E_i) = M(E_{ab}) := \begin{cases} |a| \cdot |b|, & \text{if } a \neq b, \\ |a| \cdot (|a| - 1), & \text{if } a = b. \end{cases} \tag{44}$$

Using Eq. (43), we can then easily derive the following closed-form expression for the free energy:

$$F_{\boldsymbol{\beta}} = \log Z_{\boldsymbol{\beta}} = \log \prod_{i \in I} (1 + e^{\beta_i})^{M(E_i)} = \sum_{i \in I} M(E_i) \log (1 + e^{\beta_i}). \tag{45}$$

Deriving Eq. (45) with respect to each β_i, we can then calibrate the parameters to match the target value as described in Eq. (11). The theoretical details of the derivations of Eq. (45) are presented in Appendix B. However, these close-form parameters do not account for the fact that we are interested only in sampling from the subspace of connected graphs $\mathcal{S} = \mathcal{G}_{n,\text{conn}}$. Therefore, even for this simpler model, we need to estimate the parameters using the EE algorithm in Algorithm 1. It is insightful to compare the parameters obtained using the closed-form expression of the free energy as in Eq. (45) (hence, without taking into account the connectivity constraint) with those obtained using the EE algorithm in the subspace of connected graphs.

In Table III, we report the values of the different parameters for the two methods when G_0 is the topology of the `300 IEEE` network. Not surprisingly, enforcing connectivity encourages a less sparse solution and thus the parameter values of the edge counts are lower to account for this. Furthermore, we have noted that the number of edges in a specific edge subset seems to be correlated with the steepness of the decrease in the value of the associated parameter. Understanding exactly how hard constraints, such as connectivity, affect the parameters of an ERG density is, to the best of our knowledge, still an open problem.

We show in Table IV that, in fact, the graphs generated by the simpler model cannot, on average, capture the transitivity of the real power grids. We compare the average

TABLE III. A comparison of parameters for the simpler model in Eq. (43) obtained taking the `300 IEEE` topology as reference grid G_0, using either the closed-form expression of the free energy or the EE algorithm.

Observable	G_0 value	Free-energy β	EE β	Change (%)
e_{PP}	8	−5.68	−6.19	−9.0
e_{PL}	110	−4.84	−5.22	−7.7
e_{PI}	9	−5.33	−5.83	−9.4
e_{LL}	240	−4.45	−4.60	−3.4
e_{LI}	35	−5.05	−5.38	−6.5
e_{II}	7	−3.89	−4.59	−17.0

TABLE IV. A comparison of the average values of the topological properties of interest between the simpler model arising from Eq. (42) and the one that includes the triangles arising from Eq. (13) with respect to the real values observed for the `300 IEEE` topology.

Model	m	$\langle k_P \rangle$	$\langle k_L \rangle$	$\langle k_I \rangle$	T_1	T_2	APL	λ_2	C
Reference grid `300 IEEE`	409	1.95	3.00	2.15	34	14	9.93	0.009	0.085
ERG model without T_1 and T_2	407	1.88	3.00	2.13	2.7	0.08	6.00	0.067	0.005
ERG model with T_1 and T_2	413	1.93	3.06	2.16	35	15	6.44	0.049	0.076

results obtained with the model specification as in Eq. (42) with that obtained with the specification given in Eq. (13), once again using the `300 IEEE` topology as a reference grid G_0. The simpler model produces graphs with an average degree per bus-type close to the desired ones but none of the transitivity metrics (i.e., clustering coefficient and the related triangle counts) are consistent with the observed ones.

VI. CONCLUSIONS AND FUTURE WORK

Exponential random graphs (ERGs) are some of the most well-studied random graph models. However, to the best of our knowledge, these have never been used in the context of the generation of synthetic power grids. The advantage of using ERG models in this domain is that desirable local and global topological properties can be introduced simultaneously as soft constraints in the sampling density. The proposed methodology allows the efficient generation of large and diverse samples of synthetic grids starting from a real power grid given as input. In this work, we have introduced the ERG mathematical framework, presented an approach to tackle some crucial technical issues, including a general estimation procedure for the parameters which allows for even more flexibility while using the ERG models, and presented some numerical results for synthetic power-grid generation.

Since only a single reference grid is needed as input to generate an ensemble of weakly correlated topologies that share, on average, the desired properties, our method can be efficiently used to produce a large collection of realistic grids. These grids in turn, following the rationale of the power grids from a network-of-networks perspective proposed in Ref. [32], can be used as "fragments," i.e., building blocks to be reassembled to generate larger and more heterogeneous grids, as done in Refs. [3–5]. This approach could be used in an iterative fashion, using reassembled grids as a starting point of reference for Algorithm 1.

Possible future research directions include investigating the dynamics of the chain generated in Algorithm 1 more in depth so that we can identify how well our method scales as the size of the network increases. This would help us to identify the mixing time, and thus an appropriate burn-in time t_B, in a more theoretically sound way.

An alternative approach would be to use a different chain to sample the ERG distribution. One promising example is the chain proposed in Ref. [78] to study the distribution of disordered spin systems.

As mentioned in Sec. I, the Boltzmann formulation of the ERG model allows for a straightforward form of compositional modeling, especially in the context of compositional generation using energy-based models [2]. We consider particularly interesting the possibility of using an ERG model tailored to a specific generation task, such as the one presented in this paper, as a form of guidance or conditioning for a deep-generative energy-based model, in a fashion similar to the work in Ref. [1].

Finally, since our scheme is rather general, it would be interesting to investigate its application in contexts different from the analysis of power grids.

ACKNOWLEDGMENTS

This study was carried out within the Projects of the National Recovery and Resilience Plan (PNRR) research activities of the Interconnected Nord-Est Innovation Ecosystem (i-NEST) consortium, funded by the European Union Next-GenerationEU (PNRR, Missione 4 Componente 2, Investimento 1.5—D.D. 1058 23/06/2022, ECS_00000043).

APPENDIX A: LIST OF ANALYZED GRIDS

The transmission grids used in this study are those collected and described in Ref. [50], which have been made available in a MATPOWER *testcase* format [79] in the POWER-GRID-LIB library [80] and that we have parsed using the PGLIB-OPF-PYPARSER library [81]. For each available grid, we derive a simple, undirected, and unweighted graph object. The bus types have been inferred as follows: the generators are retrieved directly from the generator list available in the MATPOWER file, the interconnections are the buses with no power generation or demand, while the remaining nodes have all been labeled as loads. In Table V, we present some topological properties used during the analysis for all connected power grids.

APPENDIX B: PROOF OF THE CLOSED-FORM EXPRESSION OF THE SIMPLER MODEL

Theorem 2. Consider a simple undirected unweighted graph $G = (V, E)$, and let $A = \left(A_{ij}\right)$ be the symmetric

TABLE V. The investigated properties of the analyzed grids from Ref. [50]. The properties reported in the table are those described in Secs. II B and II C.

Name	Number of buses	Number of branches	$\langle k \rangle$	$\langle k_P \rangle$	$\langle k_L \rangle$	$\langle k_I \rangle$	P (%)	C (%)	I (%)	Number of triangles	Number of 2-triangles	λ_2	C	APL
case30_as	30	41	2.73	2	2.77	4.5	20	73	7	6	0	0.212	0.235	3.306
case30_ieee	30	41	2.73	2	2.77	4.5	20	73	7	6	0	0.212	0.235	3.306
case39_epri	39	46	2.3	1.1	2.79	0	25	72	2	1	0	0.076	0.038	4.749
case57_ieee	57	78	2.74	3.86	2.62	2	12	82	5	9	2	0.088	0.122	4.954
case73_ieee_rts	73	108	2.96	2.88	3.11	2	45	51	4	3	0	0.04	0.025	5.983
case118_ieee	118	179	3.03	3.56	2.5	3.25	46	47	7	23	11	0.027	0.165	6.309
case179_goc	179	222	2.48	1	2.45	3.65	16	61	22	19	13	0.007	0.089	12.382
case200_activ	200	245	2.45	1	2.91	3.25	24	74	2	13	4	0.023	0.037	8.223
case240_pserc	240	348	2.89	1	3.44	0	22	78	0	49	33	0.017	0.114	8.824
case300_ieee	300	409	2.73	1.96	3.06	2.15	23	68	9	34	14	0.009	0.086	9.935
case500_goc	500	651	2.6	1.38	3.22	2.1	30	64	6	52	24	0.007	0.061	9.75
case588_sdet	588	677	2.3	2.54	2.18	2.87	21	72	7	7	0	0.004	0.011	13.495
case793_goc	793	904	2.28	2.61	2.15	2.48	22	70	8	9	0	0.003	0.01	15.331
case1354_pegase	1354	1710	2.53	2.58	1.08	2.53	19	1	80	87	14	0.005	0.056	11.151
case2000_goc	2000	2810	2.81	1.15	3.08	2.78	13	81	6	232	129	0.001	0.063	16.363
case2312_goc	2312	2830	2.45	2.01	2.51	2.71	18	70	12	52	11	0.004	0.017	15.009
case2383wp_k	2383	2886	2.42	3.01	2.33	0	14	86	0	26	3	0.003	0.009	12.759
case2736sp_k	2736	3495	2.55	3.44	2.45	3	10	90	0	56	5	0.003	0.014	13.399
case2737sop_k	2737	3497	2.56	3.56	2.45	3.5	9	91	0	57	5	0.003	0.014	13.397
case2742_goc	2742	4005	2.92	3.67	2.91	2.07	2	98	1	152	8	0.003	0.033	15.979
case2746wop_k	2746	3505	2.55	3.2	2.45	2.8	14	86	0	59	5	0.003	0.014	13.317
case2746wp_k	2746	3505	2.55	3.16	2.45	0	14	86	0	58	5	0.003	0.014	13.302
case3012wp_k	3012	3566	2.37	2.96	2.29	1.22	12	88	0	24	1	0.002	0.01	14.529
case3120sp_k	3120	3684	2.36	2.92	2.3	1.22	11	89	0	25	1	0.003	0.009	14.262
case3970_goc	3970	5712	2.88	3.34	2.86	2.54	3	97	0	156	1	0.002	0.026	17.206
case4020_goc	4020	6089	3.03	3.74	3.02	2.49	2	97	1	248	3	0.003	0.038	14.679
case4601_goc	4601	6305	2.74	3.33	2.72	3.24	3	97	0	117	1	0.001	0.017	17.409
case4619_goc	4619	7337	3.18	3.78	3.19	2.5	3	93	4	492	50	0.001	0.067	18.015
case4661_sdet	4661	5751	2.47	2.45	2.43	2.78	20	70	10	92	12	0.004	0.019	15.671
case4837_goc	4837	6622	2.74	3.22	2.72	2.54	3	96	1	250	3	0.001	0.048	23.93
case4917_goc	4917	6187	2.52	1.85	2.71	2.85	25	65	10	240	147	0.001	0.035	21.466
case6468_rte__api	6468	8065	2.49	1.83	2.6	3.27	15	84	1	351	79	0.002	0.051	14.961
case6468_rte	6468	8065	2.49	1.83	2.6	3.27	15	84	1	351	79	0.002	0.051	14.961
case6470_rte	6470	8066	2.49	1.81	2.59	4.82	15	84	1	352	80	0.002	0.052	14.985
case6495_rte	6495	8084	2.49	1.78	2.56	5.95	16	83	1	352	80	0.002	0.052	14.952
case6515_rte	6515	8104	2.49	1.77	2.57	5.9	16	83	1	352	80	0.002	0.051	14.952
case9591_goc	9591	14 042	2.93	3.74	2.92	2.43	2	97	1	557	30	0.001	0.034	17.107
case10000_goc	10 000	12 742	2.55	1.33	2.69	3.57	14	81	5	606	334	0.001	0.031	23.273
case10480_goc	10 480	16 107	3.07	3.74	3.05	3.02	3	94	2	840	81	0.001	0.053	18.606
case19402_goc	19 402	29 751	3.07	3.75	2.99	3.46	2	85	13	1152	89	0	0.04	20.883
case24464_goc	24 464	34 693	2.84	3.33	2.9	2.8	3	21	76	1346	102	0	0.048	35.994
case30000_goc	30 000	35 233	2.35	1.54	2.36	3.75	8	89	4	824	448	0	0.016	34.085

adjacency matrix associated with G. Assume that a partition $E_1, \ldots, E_K$ of all possible node pairs $V \times V$ is given, and let $A_1, \ldots, A_K$ be the corresponding block decomposition of the adjacency matrix A such that $A_k = \left(A_{ij}\right)_{(i,j)\in E_k}$. Consider the ERG model on $G \in \mathcal{G}_n$ with Hamiltonian

$$\mathcal{H}_{\boldsymbol{\beta}}(G) = \sum_{k=1}^{K} \beta_k |E_k(G)|, \quad G \in \mathcal{G}_n, \tag{B1}$$

where $|E_k(G)|$ indicates the number of nonzero entries in the submatrix A_k for the graph G, i.e., how many edges from the pairs in the subset E_k graph G actually has. Then, the partition function $Z_{\boldsymbol{\beta}}$ associated with $\mathcal{H}_{\boldsymbol{\beta}}$ is of the form

$$Z_{\boldsymbol{\beta}} = \prod_{k=1}^{K} (1 + e^{\beta_k})^{M(E_k)}, \tag{B2}$$

where $M(E_k)$ is the total number of entries in the submatrix A_k.

Proof. In view of the structure of the Hamiltonian given in Eq. (B1), the corresponding partition function is

$$Z_{\boldsymbol{\beta}} = \sum_{G\in\mathcal{G}_n} e^{\sum_{k=1}^{K}\beta_k|E_k|}. \tag{B3}$$

Since we sum over all possible graphs $G \in \mathcal{G}_n$, each block A_k of the adjacency matrix can be considered as an independent matrix and E_k represents the number of edges in the portion of the graph associated with A_k. Thus, we can rewrite Eq. (B3) as

$$Z_{\boldsymbol{\beta}} = \sum_{G\in\mathcal{G}_n}\prod_{k=1}^{K}\prod_{A_{ij}\in A_k} e^{\beta_k A_{ij}}.$$

Note that since we consider undirected and unweighted graphs, the entries of the adjacency matrix can only take values 0 or 1, i.e., $A_{ij} \in \{0,1\}$ for all i,j. As described in Ref. [56], since all considered observables are functions of the entries A_{ij} of the adjacency matrix A, we can sum over all possible graphs $G \in \mathcal{G}_n$ by summing over all possible combinations of values for each A_{ij}. By doing so, we obtain

$$Z_{\boldsymbol{\beta}} = \prod_{k=1}^{K}\prod_{A_{ij}\in A_k}(1+e^{\beta_k}) = \prod_{k=1}^{K}(1+e^{\beta_k})^{M(E_k)}.$$

■

[1] Y. Du, C. Durkan, R. Strudel, J. B. Tenenbaum, S. Dieleman, R. Fergus, J. Sohl-Dickstein, A. Doucet, and W. Grathwohl, Reduce, reuse, recycle: Compositional generation with energy-based diffusion models and MCMC, ArXiv:2302.11552.

[2] G. Hinton, Training products of experts by minimizing contrastive divergence, Neural Comput. **14**, 1771 (2002).

[3] S. J. Young, Y. Makarov, R. Diao, R. Fan, R. Huang, J. OrBrien, M. Halappanavar, M. Vallem, and Z. H. Huang, in *2018 IEEE Power & Energy Society General Meeting (PESGM)* (IEEE, 2018).

[4] S. J. Young, Y. Makarov, R. Diao, M. Halappanavar, M. Vallem, R. Fan, R. Huang, J. OrBrien, and Z. H. Huang, in *2018 IEEE Power & Energy Society General Meeting (PESGM)* (IEEE, 2018).

[5] Z. Huang, R. Huang, Y. V. Makarov, S. J. Young, R. Fan, A. Tbaileh, Z. Hou, J. O'Brien, J. C. Fuller, J. Hansen, and L. D. Marinovici, Sustainable Data Evolution Technology (SDET) for Power Grid Optimization (Final Report), Tech. Rep. (2018).

[6] J. M. Snodgrass, *Tractable Algorithms for Constructing Electric Power Network Models* (The University of Wisconsin-Madison, 2021).

[7] M. H. Mohammadi and K. Saleh, Synthetic benchmarks for power systems, IEEE Access **9**, 162706 (2021).

[8] D. J. Watts and S. H. Strogatz, Collective dynamics of "small-world" networks, Nature **393**, 440 (1998).

[9] Z. Wang, A. Scaglione, and R. J. Thomas, Generating statistically correct random topologies for testing smart grid communication and control networks, IEEE Trans. Smart Grid **1**, 28 (2010).

[10] Z. Wang, S. H. Elyas, and R. J. Thomas, in *2015 IEEE Eindhoven PowerTech* (IEEE, 2015).

[11] Z. Wang and S. H. Elyas, in *Proceedings of the 50th Hawaii International Conference on System Sciences (2017)* (Hawaii International Conference on System Sciences, 2017).

[12] Z. Wang, M. H. Athari, and S. H. Elyas, in *2018 IEEE Power & Energy Society General Meeting (PESGM)* (2018), p. 1.

[13] F. Chung and L. Lu, Connected components in random graphs with given expected degree sequences, Ann. Comb. **6**, 125 (2002).

[14] F. Chung and L. Lu, The average distance in a random graph with given expected degree, Proc. Natl. Acad. Sci. USA **99**, 15879 (2002).

[15] F. Chung, L. Lu, and V. Vu, The spectra of random graphs with given expected degrees, Int. Math. **1**, 257 (2003).

[16] S. G. Aksoy, E. Purvine, E. Cotilla-Sanchez, M. Halappanavar, and R. Lambiotte, A generative graph model for electrical infrastructure networks, J. Complex Netw. **7**, 128 (2018).

[17] O. Boyaci, M. R. Narimani, K. Davis, and E. Serpedin, in *2022 IEEE Texas Power and Energy Conference (TPEC)* (2022), p. 1.

[18] S. Pahwa, C. Scoglio, and A. Scala, Abruptness of cascade failures in power grids, Sci. Rep. **4**, 3694 (2014).

[19] M. Molloy and B. Reed, A critical point for random graphs with a given degree sequence, Random Struct. Alg. **6**, 161 (1995).

[20] M. Shahraeini, Modified Erdős-Rényi random graph model for generating synthetic power grids, IEEE Syst. J. **18**, 96 (2024).

[21] R. Atat, M. Ismail, and E. Serpedin, in *2023 IEEE International Smart Cities Conference (ISC2)* (2023), p. 1.

[22] L. Lovász, *Large Networks and Graph Limits*, Colloquium Publications (American Mathematical Society, 2012), Vol. 60.

[23] J. Nešetřil, E. Milková, and H. Nešetřilová, Otakar Borůvka on minimum spanning tree problem translation of both the 1926 papers, comments, history, Discrete Math. **233**, 3 (2001).

[24] P. Schultz, J. Heitzig, and J. Kurths, A random growth model for power grids and other spatially embedded infrastructure networks, Eur. Phys. J. Spec. Top. **223**, 2593 (2014).

[25] S. Soltan and G. Zussman, in *2016 IEEE Power and Energy Society General Meeting (PESGM)*.

[26] S. Soltan, A. Loh, and G. Zussman, A learning-based method for generating synthetic power grids, IEEE Syst. J. **13**, 625 (2019).

[27] J. Rantaniemi, J. Jääskeläinen, J. Lassila, and S. Honkapuro, A study on the impact of distance-based value loss on transmission network power flow using synthetic networks, Energies **15**, 423 (2022).

[28] H. Li, J. L. Wert, A. B. Birchfield, T. J. Overbye, T. G. S. Roman, C. M. Domingo, F. E. P. Marcos, P. D. Martinez, T. Elgindy, and B. Palmintier, Building highly detailed synthetic electric grid data sets for combined transmission and distribution systems, IEEE Open Access J. Power Energy **7**, 478 (2020).

[29] A. B. Birchfield, T. Xu, K. M. Gegner, K. S. Shetye, and T. J. Overbye, Grid structural characteristics as validation criteria for synthetic networks, IEEE Trans. Power Syst. **32**, 3258 (2017).

[30] A. B. Birchfield, E. Schweitzer, M. H. Athari, T. Xu, T. J. Overbye, A. Scaglione, and Z. Wang, A metric-based validation process to assess the realism of synthetic power grids, Energies **10**, 1233 (2017).

[31] R. Espejo, S. Lumbreras, and A. Ramos, A complex-network approach to the generation of synthetic power transmission networks, IEEE Syst. J. **13**, 3050 (2019).

[32] M. Halappanavar, E. Cotilla-Sanchez, E. Hogan, D. Duncan, Zhenyu Huang, and P. D. H. Hines, A network-of-networks model for electrical infrastructure networks, ArXiv:1512.01436.

[33] F. Arrano-Vargas and G. Konstantinou, in *2021 IEEE PES Innovative Smart Grid Technologies—Asia (ISGT Asia)* (IEEE, 2021).

[34] M. Khodayar, J. Wang, and Z. Wang, Deep generative graph distribution learning for synthetic power grids, ArXiv:1901.09674.

[35] Z. Wang, R. J. Thomas, and A. Scaglione, in *Proceedings of the 41st Annual Hawaii International Conference on System Sciences (HICSS 2008)* (IEEE, 2008).

[36] A. Birchfield and T. Overbye, in *Proceedings of the Annual Hawaii International Conference on System Sciences* (Hawaii International Conference on System Sciences, 2020).

[37] S. H. Elyas and Z. Wang, Improved synthetic power grid modeling with correlated bus type assignments, IEEE Trans. Power Syst. **32**, 3391 (2015).

[38] H. Sadeghian and Z. Wang, AutoSynGrid: A MATLAB-based toolkit for automatic generation of synthetic power grids, Int. J. Electr. Power Energy Syst. **118**, 105757 (2020).

[39] F. R. K. Chung, *Spectral Graph Theory* (American Mathematical Society, 1997).

[40] M. Fiedler, Laplacian of graphs and algebraic connectivity, Banach Center Publ. **25**, 57 (1989).

[41] H. Ronellenfitsch, D. Manik, J. Hörsch, T. Brown, and D. Witthaut, Dual theory of transmission line outages, IEEE Trans. Power Syst. **32**, 4060 (2017).

[42] F. Kaiser and D. Witthaut, Topological theory of resilience and failure spreading in flow networks, Phys. Rev. Res. **3**, 023161 (2021).

[43] A. Zocca, C. Liang, L. Guo, S. H. Low, and A. Wierman, A spectral representation of power systems with applications to adaptive grid partitioning and cascading failure localization, ArXiv:2105.05234.

[44] L. Guo, C. Liang, A. Zocca, S. H. Low, and A. Wierman, Line failure localization of power networks part I: Non-cut outages, IEEE Trans. Power Syst. **36**, 4140 (2021).

[45] L. Guo, C. Liang, A. Zocca, S. H. Low, and A. Wierman, Line failure localization of power networks part II: Cut set outages, IEEE Trans. Power Syst. **36**, 4152 (2021).

[46] F. Kaiser, V. Latora, and D. Witthaut, Network isolators inhibit failure spreading in complex networks, Nat. Commun. **12** (2021).

[47] L. Lan and A. Zocca, Mixed-integer linear programming approaches for tree partitioning of power networks, ArXiv:2110.07000v2.

[48] P. Hines, S. Blumsack, E. C. Sanchez, and C. Barrows, in *2010 43rd Hawaii International Conference on System Sciences* (2010), p. 1.

[49] T. M. J. Fruchterman and E. M. Reingold, Graph drawing by force-directed placement, Softw.: Pract. Exp. **21**, 1129 (1991).

[50] S. Babaeinejadsarookolaee, *et al.*, The power grid library for benchmarking ac optimal power flow algorithms, ArXiv:1908.02788.

[51] G. A. Pagani and M. Aiello, The power grid as a complex network: A survey, Physica A **392**, 2688 (2013).

[52] Z. Wang and R. J. Thomas, in *2015 48th Hawaii International Conference on System Sciences*.

[53] R. Albert and A.-L. Barabási, Statistical mechanics of complex networks, Rev. Mod. Phys. **74**, 47 (2002).

[54] E. Cotilla-Sanchez, P. D. H. Hines, C. Barrows, and S. Blumsack, Comparing the topological and electrical structure of the North American electric power infrastructure, IEEE Syst. J. **6**, 616 (2012).

[55] B. Cloteaux, in *2013 IEEE 2nd Network Science Workshop (NSW)*

[56] A. Fronczak, Exponential random graph models, ArXiv: 1210.7828.

[57] D. Strauss, On a general class of models for interaction, SIAM Rev. **28**, 513 (1986).

[58] S. Chatterjee and P. Diaconis, Estimating and understanding exponential random graph models, Ann. Stat. **41**, 2428 (2013).

[59] T. A. B. Snijders, P. E. Pattison, G. L. Robins, and M. S. Handcock, New specifications for exponential random graph models, Sociol. Methodol. **36**, 99 (2006).

[60] M. Talagrand, Spin Glasses: A Challenge for Mathematicians; Cavity and Mean Field Models/Michel Talagrand. eng. Ergebnisse der Mathematik und ihrer Grenzgebiete, 3. Folge, v. 46 (Springer, Berlin, 2003).

[61] S. Chatterjee and P. S. Dey, Applications of Stein's method for concentration inequalities, Ann. Probabil. **38**, 2443 (2010).

[62] T. Snijders, Markov chain Monte Carlo estimation of exponential random graph models, J. Soc. Struct. **3** (2002).

[63] N. Metropolis, A. W. Rosenbluth, M. N. Rosenbluth, A. H. Teller, and E. Teller, Equation of state calculations by fast computing machines, J. Chem. Phys. **21**, 1087 (1953).

[64] C. J. Geyer, in *Computing Science and Statistics: Proceedings of 23rd Symposium on the Interface Interface*

Foundation, Fairfax Station, 1991 (Interface Foundation of North America, 1991), p. 156.
[65] S. Bhamidi, G. Bresler, and A. Sly, Mixing time of exponential random graphs, Ann. Appl. Probab. **21**, 2146 (2011).
[66] J. Park and M. Haran, Bayesian inference in the presence of intractable normalizing functions, J. Am. Stat. Assoc. **113**, 1372 (2018).
[67] D. R. Hunter, M. S. Handcock, C. T. Butts, S. M. Goodreau, and M. Morris, ERGM: A package to fit, simulate and diagnose exponential-family models for networks, J. Stat. Softw. **24**, 1 (2008).
[68] P. N. Krivitsky, D. R. Hunter, M. Morris, and C. Klumb, ERGM 4: Computational improvements, ArXiv:2203.08198.
[69] R Core Team, *R: A Language and Environment for Statistical Computing* (R Foundation for Statistical Computing, Vienna, Austria, 2022).
[70] J. Besag, Statistical analysis of non-lattice data, J. R. Stat. Soc. D **24**, 179 (1975).
[71] M. Byshkin, A. Stivala, A. Mira, G. Robins, and A. Lomi, Fast maximum likelihood estimation via equilibrium expectation for large network data, Sci. Rep. **8** (2018).
[72] C. Gray, L. Mitchell, and M. Roughan, Generating connected random graphs, J. Complex. Netw. **7**, 896 (2019).
[73] D. M. Ceperley and M. Dewing, The penalty method for random walks with uncertain energies, J. Chem. Phys. **110**, 9812 (1999).
[74] A. Borisenko, M. Byshkin, and A. Lomi, A simple algorithm for scalable Monte Carlo inference, ArXiv:1901.00533.
[75] Code: https://github.com/franzjack/power_grids_data.
[76] H. Sadeghian, S. H. Elyas, and Z. Wang, in *2018 IEEE Power & Energy Society General Meeting (PESGM)* (2018), p. 1.
[77] M. H. Athari and Z. Wang, Introducing voltage-level dependent parameters to synthetic grid electrical topology, IEEE Trans. Smart Grid **10**, 4048 (2019).
[78] B. Scoppola and A. Troiani, Gaussian mean field lattice gas, J. Stat. Phys. **170**, 1161 (2018).
[79] R. D. Zimmerman and C. E. Murillo-Sánchez, MATPOWER user's manual (2020).
[80] C. Coffrin, POWER-GRID-LIB, https://github.com/power-grid-lib/pglib-opf (2023).
[81] L. Lan, PGLIB-OPF-PYPARSER, https://github.com/leonlan/pglib-opf-pyparser.git (2022).