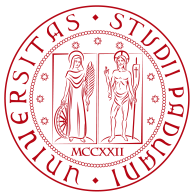


# University of Padova

Department of Mathematics "Tullio Levi-Civita"

*Master Thesis in Data Science*


# Generating Synthetic Power Grids using Exponential Random Graph Models


*Supervisor*

Professor Marco Formentin

University of Padova

*Co-supervisor*

Professor Alessandro Zocca

Vrije Universiteit Amsterdam

*Master Candidate*

Francesco Giacomarra


*Academic Year*

2021-2022

Ai miei nonni Mario e Tullio, e a ciò che di più bello ci hanno lasciato: la nostra famiglia.

# Abstract

Generating synthetic power grids as a form of data augmentation is one of the most prominent approaches in recent years to overcome the substantial lack of publicly available datasets, due to the confidential nature of the information about the real power systems. In this thesis we propose a new approach to generate synthetic power networks using the Exponential Random Graphs (ERG) models. To do this we study both the topological characteristics of the power networks using graph theory and the properties of the ERG family.

For our first proposed model, we introduce a new Hamiltonian specification with a closed form expression for the partition function. The second model that we propose is a more refined version of the previous one. Since we have a more complicated specification, the closed form of the partition function is lost and thus a new method to estimate the parameters is needed. For this reason, we develop an MCMC based algorithm to estimate the parameters of an ERG with any specification and with a constraint on the space of the graphs, which we then use to generate connected ERGs. We prove some theoretical results on the convergence of this last algorithm.

These results appear to be both useful in the specific field of power system modeling and also for the Exponential Random Graph theory.

# Contents

# List of Figures

# List of Tables

# Listing of acronyms

**ERG** . . . . . Exponential Random Graph

**ERGM** . . . . . Exponential Random Graph Model

**ER** . . . . . Erdős - Rényi (model)

**APL** . . . . . Average shortest Path Length

**MCMC** . . . . . Markov-Chain Monte-Carlo

**MH** . . . . . Metropolis-Hastings

**MLE** . . . . . Maximum Likelihood Estimation

**BFS** . . . . . Breadth-First Search

**ET** . . . . . Edge-Types (model)

# 1 Introduction

Power grids are one of the fundamental infrastructures for the proper functioning of every activity in our society. To make them always work reliably and correctly it is essential to have a thorough understanding of every aspect of these networks. In a data-based world a similar task would usually be performed by extracting all data related to these structures with appropriate techniques. However, when addressing this specific problem, there is a substantial lack of high quality real data, due to the fact that information about real networks is often limited by their owner and cannot be freely accessible by the research community.

To overcome this, one of the main approach used and studied during the last decade is the generation of synthetic power grids whose features mimic the ones of the real networks using specific mathematical models. It seems natural to see electricity networks as complex graphs whose structure exhibits distinctive properties. Given that, one could theoretically rely on the existing literature about generative graph models to perform this task, but due to some specific aspects of the grids, using already existing and well-studied models has been until now unsuccessful, leading to the use of specially constructed procedures that, albeit useful, often lack sufficient rigorous analysis, limiting the possibilities of using graph-theoretical and probabilistic tools to further enhance our understanding of these special networks.

For this reasons we propose here a new approach, that uses for the first time, to the best of our knowledge, Exponential Random Graph models (ERGM) to generate synthetic power grids. Exponential Random Graphs are one of the most popular family of graph models, especially in the field of social network analysis. These models are widely studied for their good

statistical and probabilistic properties and because they can be extremely flexible with a proper specification.

In this work we provide examples of ERG model specifications that can fit some properties of the real grids. The main results of this thesis are shown in chapter 5 and include a new general ERG specification with a close form of the partition function, an ERG specification that mimic the main topological properties of a real grid and a general procedure to estimate the parameters of a wide class of ERG models using a Markov chain Monte-Carlo inspired algorithm, together with the rigorous proof of its convergence. It is worth mentioning that these results can find applications to a wider class of problems other than only synthetic power grids generation and in fact can be regarded as purely relative to Exponential Random Graph theory.

All the computational experiments done in this thesis were performed on an HP Omen 15-dc1xxx laptop with a Intel(R) Core(TM) i7 CPU, and a RAM of 16 GB.

The thesis is organized as follows:

- In Chapter 2 we do a review of the current literature about power system modeling, highlighting the proposed approaches that include the analysis of the topology of the grids and discussing their strengths and weaknesses, reporting the results they obtained compared to real grids.
- In chapter 3 we describe in depth the history of the Exponential Random Graph Models and their properties, with a focus on simulation and sampling from these models by using Markov chain Monte-Carlo methods. We also briefly cite some of the most recent literature about the study of ERG models through *graphons* theory.
- In chapter 4 we explain the parsing, preprocessing and selection procedures of the available grids in order to use them as a reference to specify and validate our models.
- Chapter 5 represents the core of this thesis, since we describe our approach as well as our main results together with their rigorous proofs and motivations. We show also how our results compare to the real grids. This chapter is organized following the chronological evolution of our models, starting from the very first model specification, for which we also prove one of the main theorem in our work, then going through the refined specifications that in the end have lead to our final model, for which we provide an in-depth analysis.
- In Chapter 6 we state the conclusions of our work and we define what are the main improvements that we could do in the future to make our approach even more accurate, and also the steps that should be included to make our grids realistic from an electrical perspective.

# 2

# Power Systems Model

In this chapter we will give an overview of the principal properties of the power networks and also we will review the current literature on the modeling of these networks, with an emphasis on the articles and papers whose approach focuses more on the topological properties of the grids. In the first Section 2.1 we will describe briefly the transmission power network model, focusing on the properties more useful during the synthetic grid generation. In Section 2.2 we do a comprehensive review of the main proposed models and procedures for generating synthetic systems, reporting the results obtained with such models by their authors. Lastly, in Section 2.3 we summarize the reasoning behind the use of ERG models to tackle this problem and what will be the advantage of using such an approach compared to the ones already present in the literature.

## 2.1 Power Grid Model

Power grids are interconnected networks that deliver electricity from producers to consumers, composed of nodes called buses connected through links called power lines. We can distinguish two major types of power networks, namely the distribution network and the transmission network. The distribution networks have shorter power lines (often referred to as distribution power lines) and serve the function of electricity transportation for short distances and low-voltage levels. The transmission network is used to transport electricity for long distances working at high voltage levels, having longer power lines (also referred to as transmission power

lines). for this thesis we will focus only on the high-voltage transmission networks. To better understand the modeling of power grids, we first introduce here some basic notations of graph theory. An undirected, unweighted graph is a pair of objects $G = (V, E)$ where $V$ is a set of vertices (also called nodes) and $E$ is a set of vertices' pairs, whose elements are also called edges or links. For each graph we can define an associated adjacency matrix $A = A(G) \in \{0, 1\}^{|V|\times|V|}$, that is a symmetric matrix for which it holds:

$$A_{i,j} = \begin{cases} 1 & \text{if}(i, j) \in E \\ 0 & \text{otherwise.} \end{cases} \tag{2.1}$$

Moreover we also define the Laplacian matrix $L$ of a graph, defined as

$$L_{i,j} := \begin{cases} \deg(v_i) & \text{if } i = j \\ -1 & \text{if } i \neq j \text{ and } v_i \text{ is adjacent to } v_j \\ 0 & \text{otherwise.} \end{cases} \tag{2.2}$$

The Laplacian matrix is useful to analyze many properties of the Graph as we will see in the following. A power network can be described as a graph where the nodes represent **buses** and the edges represent **transmission lines**. We can distinguish three types of buses:

- **Generators** ($P \subset V$) represent the components where the electricity is produced. Examples of generators are fossil-fuel power stations, nuclear power plants and Solar panels.
- **Loads** ($L \subset V$) represent the components where the electricity is consumed. Loads can represent, for example, industries, residential neighbourhoods or private houses.
- **Interconnections** ($I \subset V$) represent point of passage for the electric current, allowing for more complex transmission circuits.

In the following we will assume that $V = P \cup L \cup I$ and that each node belongs exactly to one type. In the Fig. 2.1 we show a toy example of a power grid in graph notation. The yellow node represents a generator, the green nodes represent interconnection nodes and the red ones are the loads.

From an electrical point of view, any transmission grid works in a specific way under some basic constraints [4]. Albeit this is not the main topic of this thesis, we summarize briefly here these constraints to give a better understanding overall:

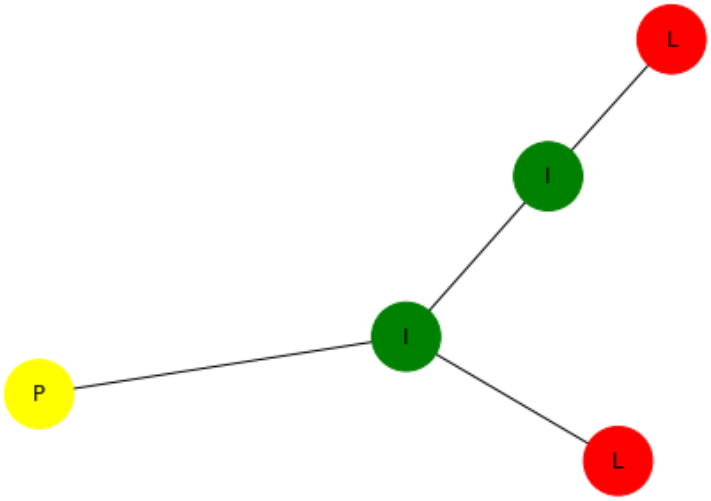


**Figure 2.1:** Toy example of a power grid with 5 nodes

- let $\mathbf{g} = [g_1, g_2, \ldots, g_{|V|}]$ be the vector of the power generated at each node in the grid, and let $\mathbf{d} = [d_1, d_2, \ldots, d_{|V|}]$ be the vector of power demand at each node. Then, at any time the total generation must be equal to the total demand, i.e.,

$$\sum_{i}^{|V|} g_i = \sum_{i}^{|V|} d_i. \tag{2.3}$$

- The amount of energy that can be generated by each node is limited, that is $g_i^{\min} < g_i < g_i^{\max}$ for each $i$.

- The energy generated flows along the transmission lines in the form of alternating current to meet the demand of each node. Each lines has a specific capacity, a maximum amount of power that is allowed to flow on it, in order to avoid overheating. High-voltage lines have an emergency shutdown system whenever the current flowing exceeds a certain threshold (often set to be smaller than the maximum capacity of the line).

Using the graph theory framework, the bus types can be seen as a nodal attribute and capacities as an edge attribute or edge-weights if we use a weighted graph. Other attributes that can be considered are the length for the transmission lines, and the geographical position for the buses.

From a topological point of view, it has been observed that power system have some peculiar properties that we will briefly describe now:

- **Connectivity**: all the power systems are connected graphs which means that there exists a path between any two nodes. This is due to the fact that the power flow is expected to be able to reach any point in the grid from any starting point. If we admit parallel edges (*i.e.,* multiple edges that connect the same pair of nodes) it has been argued that most of the grids should be 2-connected graphs, which means that the removal of any single edges does not make the graph disconnected. This last property however is often disregarded to preserve simplicity of the model, and also because not all the grids are effectively 2-connected.

- **Sparsity**: all the power systems are sparse graphs, which means that the number of edges is of the same order of magnitude as the number of nodes, formally $|E| \approx \mathcal{O}(|V|)$. This fact is quite natural given the high costs to build and maintain transmission lines, as well as the intuitive design principle prescribing transmission lines' not to intersect with each other.

- **Average Node degree**: the average node degree $\langle k \rangle$, *i.e.,* the average number of nodes each node is connected with, *i.e.,* the average of the degrees of each node $k_i$. It is defines as

$$\langle k \rangle := \frac{1}{|V|} \sum_{i}^{|V|} k_i = \frac{1}{|V|} \sum_{i}^{|V|} \sum_{j}^{|V|} A_{i,j}. \tag{2.4}$$

  It has been showed to be quite stable regardless of the network size, and it oscillates between values of 2 and 5 [5].

- **Average shortest path length**: the average path length is the average length of the shortest path between any two nodes in the graph. Let $d(v_i, v_j)$ be the shortest path in hops between any two nodes $i$ and $j$, then we define the average shortest path length (APL) as (2.5)

$$\text{APL} = \frac{2 \sum_{i,j} d(v_i, v_j)}{|V| \cdot (|V| - 1)} \tag{2.5}$$

  For power grids, it grows proportionally to $\frac{\ln(N=\text{number of nodes})}{\ln(\langle k \rangle)}$. This is consistent with the results of Albert and Barabasi [6].

- **Average Clustering Coefficient**: the local clustering coefficient of a node is defined as the ratio of the number of *triangles* (which in turn are defined as any group of three nodes with an edge between each pair, graphically forming a triangle) to which the node belongs, and the number of possible triangles that could exist between the node and its neighbours, in formulas (2.6):

$$C_i = \frac{1}{k_i(k_i - 1)} \sum_{j,l} A_{ij} A_{jl} A_{li}. \tag{2.6}$$

The average clustering coefficient is defined as the average among all the nodes of the local clustering coefficients and can also be computed directly from the adjacency matrix using the following (2.7):

$$C = \frac{\sum_{i,j,k} A_{ij} A_{jk} A_{ki}}{\sum_i k_i (k_i - 1)} \tag{2.7}$$

It has been observed that the average clustering coefficient of power networks is much higher than the one of other type of sparse graphs, effectively meaning that grids present many more triangles [7]. This could be also be due to the fact that we want 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.

- **Algebraic Connectivity**: the algebraic connectivity, $\lambda_2$, is the second smallest eigenvalue of the Laplacian matrix, also called the Fiedler eigenvalue, reflects the connectivity of the graph. In particular its value is greater than 0 if and only if the graph is connected, and its magnitude gives an idea of how well the graph is connected. For power grids it has been found that it exhibits scaling property with respect to the network size [8].

## 2.2 Synthetic Power Grid generation

The generation of synthetic grids is a way to augment the availability of public data for both the research community and industry. This approach rose in popularity in the last decade, and is thus a relatively new topic. The interpretation of a power grid as a network with attributes is both straightforward and powerful, thus most of the approaches in the literature work within this framework. The first models for synthetic grids were made with a focus on the electrical properties, neglecting the topological structure; for example in [9] the authors use a tree-like topology with a small number of nodes to study cascading failure blackouts, and in [10] the authors aim to study contingency and disturbance propagation using ring-like topologies. The crucial observation is that the purpose of these synthetic power grids was not to have realistic data per se, but to highlight some electrical properties using oversimplified models.

The first attempt that we are aware of that combines both the rigorous study of sparse complex networks, done for example in [6], with the peculiar characteristics of a power system can be found in the work of Wang and Scaglione in 2008 [5]: they propose a model to generate synthetic power grids of scalable size and random topologies with nodal locations chosen at random according to some probability distribution. In their work they still consider some classical graph model, such as the *"Smallworld"* model introduced by Watts and Strogatz [11], which

was considered by its authors as a good approximation even for electrical systems, but they highlight how these models cannot capture some of the topological properties of the power grids, especially the coexistence of connectivity and a small average node degree. Thus they express the need of specific models for the considered problems.

Within this rationale the same authors made another considerable improvement, introducing a new model that they call *RT-nested-Smallworld* [12]. This model uses nested Smallworld models to replicate the characteristics of a real grid, and is build in three different steps:

1. first it forms connected subnetworks with size limited by the connectivity requirement;
2. then the subnetworks are connected through lattice connections;
3. finally, the line impedances are generated from some specific probability distribution (depending on the network size) and assigned to the links in the topology network.

Aside from the description of the model it is worth mentioning that the authors, for the first time, examined the empirical distribution of nodal degree of the real networks, analyzing both the probability mass function and the probability-generating function of the nodal degrees per bus type, and from this they have concluded that real grids have a nodal degree distribution that corresponds to the sum of a truncated geometric distribution and an irregular discrete random variable. Moreover this analysis also proved that the nodal degree distribution differs between different bus types, an observation that will be crucial both in the future works in the literature and also for our proposed models in Chapter 5.

The *RT-nested-Smallworld* model however does not provide a way to produce a correct bus-type assignment by itself. Therefore, Wang et al. [13] propose a new measure, called *"Bus-type Entropy"*, that incorporates both bus-type ratios and link type ratios and can be used to identify the presence of correlation among the bus type assignments of a realistic grid. Using this new measure the authors improved the *RT-nested-Smallworld* by proposing an optimization algorithm to make the bus-type assignment in the synthetic grids by minimizing the distance in terms of a quantity (that they call *d-scaling* and they consider a topological property inherent of the grids) based on the Bus-Type Entropy between [14], solving one of the biggest weaknesses of their previous model.

Another work of Wang et al. [8] investigates the scaling properties of some topological and electrical properties of power grids and proposes a new specification for the Bus-Type Entropy measure that they prove to have better numerical stability. This last work provided new ways to improve and validate the synthetic grid generation using the *RT-nested-Smallworld*, and

together with the in-depth analysis of the statistical properties of the grids done by the same authors in [7] led to the development of a MATLAB-GUI toolkit called *"AutoSynGrid"* [2] for the automatic generation of synthetic power grids. This toolkit is able to generate synthetic networks with just the network size as an input, however number of branches, loading level (ratio of total active load to the total generation capacity of power grid), reference system (real grid used as a reference for properties such as generator and loads setting), Bus-Type Entropy, and generation cost modelling approach (needed to perform energy economic studies on the synthetic cases).

The networks generated with this procedure are validated by measuring how close the generated synthetic grids are to the real grid of reference with respect to some considered topological and electrical properties. Table 2.1 below describes the results obtained by the authors with respect to the number of nodes ($N$), number of edges ($E$), average degree $\langle K \rangle$, average path length (APL) and algebraic connectivity ($\lambda_2$). For the last three the authors report a interval within which each value is considered acceptable according to the respective values for the real grids.

| Networks | $N$ | $E$ | $\langle k \rangle$ | APL | $\lambda_2$ |
|---|---|---|---|---|---|
| AutoSynGrid-500 | 500 | 890 | 3.5 | 6.22 | 0.011 |
| Valid Interval | | | [2-5] | [2.5-10.5] | [0.004-0.040] |
| AutoSynGrid-1000 | 1000 | 1830 | 3.6 | 12.7 | 0.008 |
| Valid Interval | | | [2-5] | [8.5-17.5] | [0.002-0.020] |
| AutoSynGrid-3000 | 3000 | 5580 | 3.6 | 16.7 | 0.003 |
| Valid Interval | | | [2-5] | [12-20] | [0.0005-0.005] |

**Table 2.1:** Results obtained with the AutoSynGrid Toolkit by its authors in [2]

The average degree seems to be almost fixed regardless of the network size, albeit within the accepted range. The tolerance for the average path length and algebraic connectivity seems loose, but still realistic. With these intervals all the generated grids can be regarded as realistic, however we must address that the data about the average clustering coefficient of these grids are not reported.

A different approach can be found in the recent works of Soltan *et al.* [15, 16]: being provided with detailed geographical data of real power grids, they have developed methods to generate synthetic spatially embedded synthetic networks, *i.e.,* they focus on the spatial distribution of buses and lines. In addition to more common statistics (like average degree, degree distribution and also the Intersection occurrence first introduced in [17]) they propose a novel statistic,

the line length distribution similarity, measured as the Kullback-Leibler Divergence between the real life line length distribution and the synthetic ones. The generation algorithm they propose, which they have called *"Geographical Network Learner and Generator"* (GNLG) and later renamed *NIMBLE*, includes three different sub-procedures that we briefly describe here:

1. *Spatially Distributed Nodes Generator* (SDNG): given the set of nodes of the real graph $G$, a Gaussian Mixture Model is used to cluster these nodes based on their geographical proximity, finding the best number of clusters according to the Bayesian Information Criterion. Then having obtained both the categorical probability and the mean and covariance of a Gaussian distribution for each cluster, it uses these parameters to sample node locations for the synthetic grid.

2. *Tunable Weight Spanning Tree* (TWST): the nodes position generated in the last step are now connected using a tree-like structure that is said to imitate the evolution of a real grid. At each iteration $i$, it samples a node $j$ from the set of nodes that weren't already sampled according to a probability that depends on the node sampling probability obtained during the SDNG step and on a specific tuning parameter $\kappa$, then it defines a permutation of the sampled node index $\nu(i) \leftarrow j$ and it removes $j$ form the set of considered nodes in the future iterations. After this procedure, the algorithm connects each node to its nearest neighbour according to the new permutation index $\nu$.

3. *Reinforcement procedure* the aim of this last step is to make so that the generated network looks similar to the considered real one, according to some topological properties, in particular the clustering coefficient and the average path length. In order to do so the algorithm uses a rationale akin to the preferential attachment model defined in [18] that takes into account also the geographical properties of the network. The low degree nodes in high-density areas (the areas with most nodes) are randomly linked to a nearby high degree node. The probabilities used to choose the nodes are determines according to some input parameters $\alpha, \beta, \gamma, \eta$.

To test the performance of the GNLG algorithm, the authors compare the results they obtained with the real grids they used as a reference, that are the grid corresponding to a portion of the Western Interconnection (WI), one of the two major interconnections of the US, and two grids that represent two regional entities that operate under the Eastern Interconnection (EI), which is the other major interconnection, the SERC Reliability Corporation (SERC), which is as large as the WI, and the Florida Reliability Coordinating Council (FRCC), which is smaller. We report here a portion of the comparison table that they showed in [15].

| Networks | APL | C |
|---|---|---|
| $G_{\text{WI}}$ | 17.33 | 0.049 |
| $G'_{\text{WI}}$ | 17.4 | 0.045 |
| $G_{\text{SERC}}$ | 19.71 | 0.049 |
| $G'_{\text{SERC}}$ | 20.26 | 0.048 |
| $G_{\text{FRCC}}$ | 11.68 | 0.075 |
| $G'_{\text{FRCC}}$ | 11.81 | 0.045 |

**Table 2.2:** Results of the GNLG algorithm *w.r.t.* clustering coefficient (C) and average path length (APL).

All the synthetic counterparts considered statistics mimic closely the ones of reference, thus this procedure can be said to be a very solid option to create networks that copy the topology structure of the input grids.

Birchfield *et al.* [19, 17, 3] propose another modelization which focuses on the geographical properties of large grids, in a similar fashion to the work of Soltan *et al.* [15, 16]. They use also an algorithm consisting of multiple steps, which we will now outline:

1. the procedure starts by using information about the considered area geography, population and other possible electrical constraints to determine the locations of all the possible nodes.

2. A clustering technique is then used to assign nominal voltage to each node, then nodes within each cluster are connected using low-impedance lines.

3. The geographic nature of the procedure allows for considerations on the lines' length: they use the fact that researches have shown that all the lines in real power grids must be part of the set of the geographic Delaunay's triangulation of the nodes and its second and third neighbours to determine the set of candidate lines.

4. The line planning algorithm starts by using an arbitrary selection of lines among the set defined above. Then, following a procedure similar to simulated annealing, a two step sub-procedure consisting of random removal of a line and "smart" addition of a line (according to the desired properties) is iterated for each clustered sub-network until the obtained network is realistic from both a geographical and electrical point of view using the $N-1$ contingency analysis (*i.e.,* testing if the failure of a line or node would not propagate, ensuring that each component would still work if anyone fails, meaning that $N-1$ components are still available). The addition of the lines is done taking into consideration the line length (shorter lines are encouraged) and to match the distribution of Delaunay's neighbours, that is the proportion between first, second and third

neighbours along the Delaunay's graph. Sensitivity (which quantifies the impact of each candidate transmission line on the contingency robustness of the transmission system, within the DC power flow modeling) is incorporated in the line evaluation process by adding a penalty to each candidate line negatively proportional to their sensitivity values, thus encouraging the addition of lines that mitigate critical contingency overloads.

The grids generated by this algorithm are compared by the authors with large similar real nets *. This comparison is summarized by table 5.1:

| Networks | $N$ | $\langle k \rangle$ | APL | $C$ |
|---|---|---|---|---|
| EI | 36187 | 2.61 | 29.2 | 0.044 |
| Synthetic 70K | 34999 | 2.74 | 36.7 | 0.048 |
| WECC | 9398 | 2.58 | 18.9 | 0.058 |
| Synthetic 20K | 11765 | 2.99 | 22 | 0.071 |
| ERCOT | 3827 | 2.61 | 14.2 | 0.032 |
| Synthetic 5K | 2941 | 3.12 | 13.7 | 0.089 |

**Table 2.3:** Results obtained with the procedure described in [3]

The procedure shows great adaptability and scalability, being able to generate a large synthetic grid with more than 30000 nodes that retains realistic topological properties.

Another way to tackle the problem of synthetic grid generation, from a complex network theory perspective, can be found in [1]. The authors consider the current state-of-the-art models (including the cited above *RT-nested-Smallworld* introduced in [12] and the work from Birchfield *et al.* [19]) not able to represent closely the topological properties of the Power networks. In particular they investigated the degree distribution $d(x)$ of the grids and found that they can be heterogeneous, with some grids having a power-law distribution (*i.e.,* $d(x) \sim \beta x^{-\alpha}$) and others being better represented by an exponential distribution (*i.e.,* $d(x) \sim \gamma e^{-\lambda x}$). From this consideration they assess the need of a parametrical model that can generate synthetic power grids and can be adjusted with respect to different topological properties and network sizes. Thus the authors propose a model which follows the evolutionary nature of the power networks while also taking into consideration multiple topological characteristics as a way to validate the proposed evolutions of the grids. We report here the flowchart of their procedure:

*In particular the authors consider most of the large north American grids, including the Eastern Interconnection (EI), the Western Interconnection (WECC) and the Texas Interconnection (ERCOT)

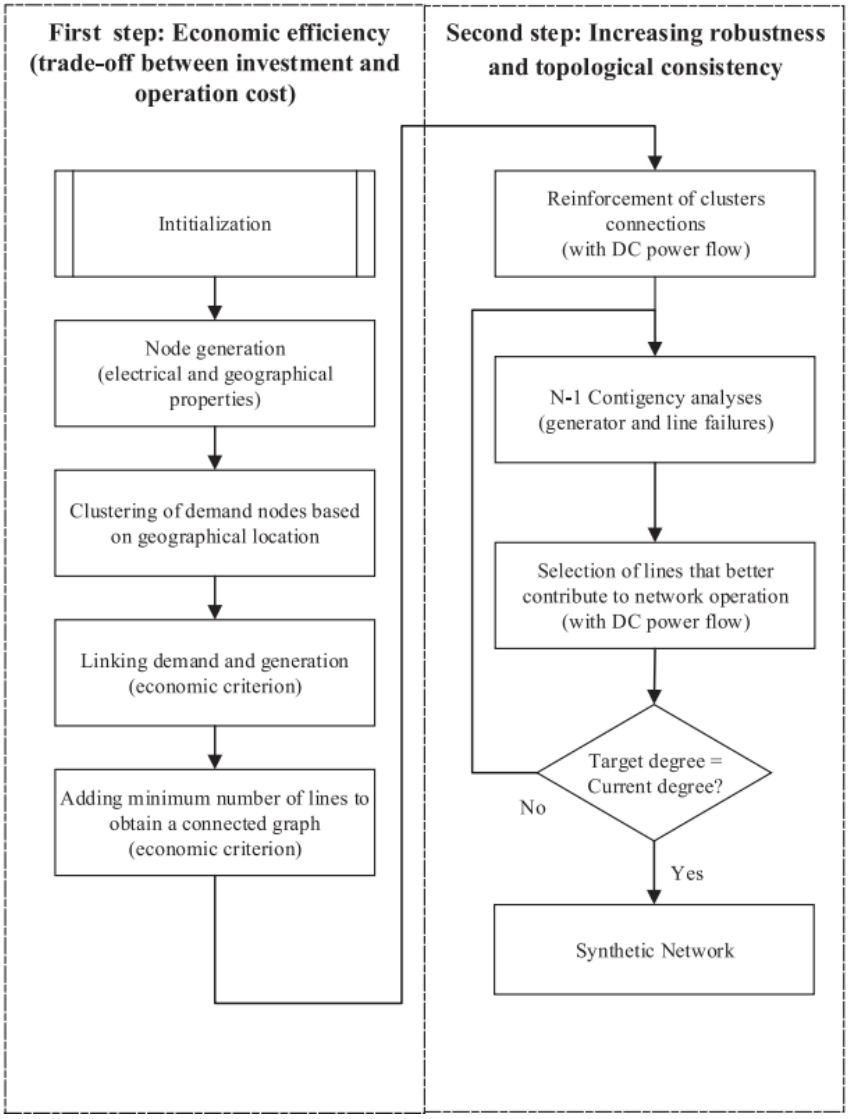


**Figure 2.2:** Flowchart of the model proposed in [1]

1. As done by Birchfield [19] and Soltan [15] the procedure starts by identifying the nodes' locations considering electrical and geographical properties, then the Loads are clustered according to their geographical locations.

2. The generators are linked with the Loads by a simple transmission network using an economic criterion: a trading process between customers (loads) and suppliers (generators) is simulated, in which each cluster of customers will try to find the best supplier that is able to satisfy its energy demand on the lowest operational cost.

3. In the next step the minimum number of lines is added to the graph in order to make

the graph connected (since different clusters might be disconnected as a result of the last step). The lines added are again the ones that ensure connectivity at the lowest operational cost.

4. After a basic topology is created, the procedure begins the reinforcement phase: the connections between clusters are improved following DC power-flow considerations (in particular if the removal of one current line would disconnect the graph, another line is added to prevent this).

5. The model checks if the topology guarantees that demand can be met even after the failure of a line or a generator using a $N - 1$ contingency analysis (*i.e.,* testing if the failure of a line or node would not propagate, ensuring that each component would still work if anyone fails, meaning that N-1 components are still available).

6. After each addition of a line the procedure checks both if the topology gives rise to a realistic power-flow and also if the current average degree of the graph is the desired one. If both conditions are met, the procedure stops and the generated network is regarded as a synthetic power grid.

We report in table 2.4 the results obtained by the authors, comparing the topological properties of the real grids with the properties of the real ones used as reference, which in this case are the high-voltage Spanish-Portuguese (SP) one and high-voltage French (Fr) one.

| Networks | $N$ | $E$ | $\langle k \rangle$ | APL | $C$ |
|---|---|---|---|---|---|
| SP real net | 304 | 434 | 2.855 | 8.886 | 0.111 |
| SP synth net | 304 | 434 | 2.855 | 9.184 | 0.105 |
| Fr real net | 217 | 283 | 2.608 | 8.279 | 0.144 |
| Fr synth net | 217 | 283 | 2.608 | 8.687 | 0.147 |

**Table 2.4:** Results obtained with the procedure described in [1]

All the generated grids mimic closely the topological characteristics of the real ones, with only the average path length showing a systematic deviation, that is negligible considering its magnitude. It is worth mentioning that the size of the considered grids is small, thus is difficult to determine how well the procedure scales with the network size.

Recently a completely different approach than the ones seen before was used by the Pacific Northwest National Laboratory to develop the SDET (*Sustainable Data Evolution Technology*) tool to create open-access synthetic grids datasets [20]. The methodology used to build the synthetic networks is summarized in [21], and consists in reassembling anonymized fragments of

real grids to build a new base topology that is then optimized to mimic the considered properties of a real grid. Furthermore, this methodology is justified by the authors by doing a topological analysis of the real power grids from a network of networks perspective [22, 23], that is stated to highlight new structural characteristics of power systems. In particular the authors in [22] analyze sub-networks with the same voltage levels as single graphs and then the interconnection of these sub-graphs as a new graph itself, investigating in both cases typical topological properties considered in the power system analysis (see Section 2.1 above) such as average shortest path length and clustering coefficient. The procedure needs to have a collection of real fragments that are collected and anonymized from real grids; the fragments are obtained by using several techniques to cluster the nodes real world networks into small node subsets, then the existing topology for each of these node subsets is corrected by replacing any link to other clusters' nodes with a link to a node within the considered cluster. This method allows to generate hundreds of fragments even from a medium size grid ($\sim 3000$ nodes). In order to be able to rank and select the fragments obtained, the authors define two quantities in the following way: Let $\mathcal{E}$ be a set of types $\rceil_\infty, \rceil_\in, \ldots$ of electrical properties such as generators,loads,lines, etc. and let $\mathcal{F}$ be the set of fragments. For each $e \in \mathcal{E}$ let $g_e$ be the desired number of elements of a specific type, defined by the user of the procedure. For each $f \in \mathcal{F}, e \in \mathcal{E}$, let now $x_{f,e}$ be the number of elements of type e present in fragment $f$. Then for a specific collection of fragments $\{c_f\}_{f\in\mathcal{F}}$ we define

$$\left| g_e - \sum_{f\in\mathcal{F}} c_f x_{f,e} \right| < \epsilon g_e, \tag{2.8}$$

as the global error with respect to $e \in \mathcal{E}$, with $\epsilon > 0$ being a control parameter. In a similar way we define the $L^2$ error (up to a scaling) of a single fragment $f$ as

$$\sum_{e\in\mathcal{E}} \left( \frac{x_{f,e}}{n_f} - \frac{g_e}{n} \right)^2, \tag{2.9}$$

with $n_f$ being the number of buses in fragment $f$ and $n$ the desired number of buses of the final grid defined by the user. After collecting enough fragments the procedure to generate synthetic grids follows the steps that we summarize here (we omit here for sake of brevity and since is beyond the scope of this thesis some of the electrical details of this method. For more information, refer to this link):

1. To begin with some general desired properties are given as an input by the user. A single

fragment is then chosen as an initial kernel. This choice is made by selecting the fragment that minimizes (2.9)

2. The subsequent fragments are selected from the available set in an iterative fashion with the goal of creating a collection able to satisfy (2.8) for each $e \in \mathcal{E}$ and with the minimum $\epsilon$. Starting from the second fragment, the next fragment is chosen randomly among the ones that share similar boundary buses (the buses located at the limits of the net) with the fragment chosen at the iteration before and are able to satisfy the condition given by (2.8) with the current choice of $\epsilon$. If no fragment satisfies this latter condition, then it is relaxed by increasing the value of $\epsilon$.

3. After forming a collection that satisfies the imposed conditions, a geographic structure is imposed to the fragments in order to satisfy two major properties of the net: $N-1$ robustness and planarity of the graph. Given this constraints, the fragments are then connected with each other iteratively by forming additional edges referred to as "connectors". After each connector is added to the topology, the impedance of the tie-lines is reduced and then Optimal Power flow analysis and a $N-1$ robustness analysis are performed to check the reliability of the system.

4. If the $N-1$ robustness is satisfied and the OPF is solved, the method checks the connectivity of the network. If the network is connected the procedure stops and the synthetic grid given as an output, otherwise the step before is repeated.

As stated in the procedure, this method highly relies on the available fragments in the library and on the properties given as an input. We report in Table 2.5 the results obtained when analyzing the grids generated by this method.

| Networks | $N$ | $E$ | $\langle k \rangle$ | APL | $C$ |
|---|---|---|---|---|---|
| SDET 588 | 588 | 686 | 2.302 | 13.49 | 0.01 |
| SDET 2312 | 2312 | 3013 | 2.448 | 15.008 | 0.017 |
| SDET 2853 | 2853 | 3921 | 2.548 | 16.53 | 0.046 |
| SDET 4661 | 4661 | 5997 | 2.467 | 15.67 | 0.018 |

**Table 2.5:** Results obtained with the SDET procedure

The synthetic networks seem reliable with respect to the considered topological properties and the idea of a method to reassemble pieces of existing topologies seems promising, even if it is limited by the available real grids used to form the fragments' library. As we will state in the conclusions, we will propose to apply a similar rationale also for our approach when dealing with bigger grids.

## 2.3 Remarks on the current models

As we have seen in Section 2.2, most of the generative models proposed in the literature are build using a multiple step procedure, with the rationale that a step-like structure can reproduce better the real evolution of a grid. Furthermore, topological and electrical properties are often considered in separate steps when generating a synthetic grid.

The models that we are going to propose in Chapter 5 use a completely different approach. In fact, we focused on the generation of realistic topologies with the least amount of input data, neglecting for instance geographical properties or line lengths. We believe that, given a realistic simple unweighted network, the other layers of a realistic grid, such as the power-flow and geographical attributes, can be added later in subsequent procedures that are beyond the scope of this thesis.

Moreover the models that we propose in this thesis belong to the Exponential Random Graph (ERG) family (which will be introduced in Chapter 3) and this allows for a better statistical and probabilistic analysis of the generated grids, highly benefiting not only our current work but also any further development of these models.

# 3

# Exponential Random Graphs

This chapter aims to give more details about the history, motivations and properties of Exponential Random Graphs (ERG). The exposition roughly follows the comprehensive overview written by Agata Fronczak [24].

In the first section 3.1 we will review the historical background of these models, in the second section 3.2 we will define and motivate the model and highlight the main properties and finally in the last section 3.3 we will discuss the simulation and estimation of Exponential Random Graph models using Markov chain Monte Carlo methods and also some of the possible problems arising when using these techniques.

## 3.1 Historical Background

The first general random graph ensemble was defined in [25] with the aim of exploring biological networks by Solmonoff and Rapoport in 1951, who considered the set of all simple undirected graphs with a fixed number of nodes $N$, and for each node-pair an edge was connecting the two with probability $p$. This model was later extensively studied and popularized by Erdos and Renyi [26], thus being known from then onward as the Bernoulli model or Erdos-Rényi model (ER-model). Albeit not yet formalized in that way, the ER-model is the first example of an Exponential Random Graph model (specifically, as we will understand better in the following, corresponds to an ERG whose Hamiltonian considers only the number of Edges as an observable).

The first exponential family probability distribution for random graphs was proposed in the early 1980s by Holland and Leinhardt [27], and then a more general definition similar to the one used nowadays was given by Frank and Strauss [28].

## 3.2 Model Definition and properties

We want to build the ensemble model $\mathcal{G} = \{G\}$ that we assume to be the underlying model from which we would sample some graphs (in practice often we start from a real-world network and we want to build the ensemble of this specific graph realization). We assume that some measurable properties of the graphs should be shared on average across all the ensemble. We call those properties the *graph observables*. Typical examples of observables used in ERG models are functions of sub-graphs counts: if we move in a space of graphs with $N$ nodes we can define as a sub-graph any graph structure that includes $m < N$ nodes (in practice often we have $m << N$). Common sub-graphs considered in the literature are single edges, 2-paths and triangles Fig. 3.1.

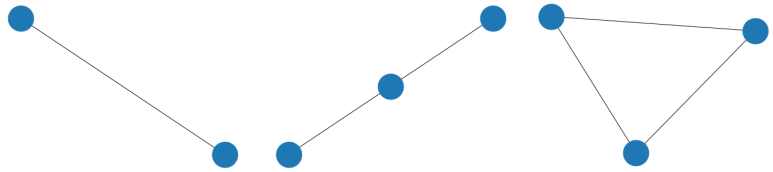

**Figure 3.1:** Sub-graphs: from left to right, single edge, 2-paths and triangle

It seems then reasonable to assume that the observables $x_1(G), x_2(G), \ldots,$ of a graph $G$ determine the probability of observing that graph as the network realization. In the ERG model we assume that the probability of observing a certain graph $G$ with observables $x_1(G), x_2(G), \ldots, x_r(G)$ is given by

$$P(G) = \frac{e^{H(G)}}{Z}, \tag{3.1}$$

where $H(G)$ is called the *Hamiltonian* of the model and takes the following form

$$H(G) := \beta_1 x_1(G) + \beta_2 x_2(G) + \cdots + \beta_r x_r(G), \tag{3.2}$$

with $\beta_1, \beta_2, \ldots, \beta_r$ being parameters of the model and $Z$ takes the name of *partition function*,

which can be calculated from normalization condition

$$\sum_{G\in\mathcal{G}} P(G) = \frac{1}{Z}\sum_{G\in\mathcal{G}} e^{H(G)} = 1, \tag{3.3}$$

thus implying

$$Z = \sum_{G\in\mathcal{G}} e^{H(G)}. \tag{3.4}$$

Now notice that for each observables the follow equation holds

$$x_i^* = \langle x_i\rangle = \sum_{G\in\mathcal{G}} x_i(G)P(G) \quad \forall i \tag{3.5}$$

thus, the parameters $\beta_1, \beta_2, \ldots, \beta_r$ should be calculated, either analytically or numerically, from equation (3.5) after fixing the values of $x_i^*$, for each $i$.

### 3.2.1 Main properties

We will now discuss a more general statement for (3.5). Consider any quantity $y = y(G)$ depending only on the graph $G$. We can calculate an estimate of this quantity as an average over the whole ensemble by using

$$\hat{y} = \langle y\rangle = \sum_{G\in\mathcal{G}} y(G)P(G) = \frac{1}{Z}\sum_{G\in\mathcal{G}} y(G)e^{H(G)}. \tag{3.6}$$

This means that we can infer general properties of the, supposed, underlying model that generated the observed network using other properties (that maybe we are assuming to be true and shared across all the graphs) as the observables. For example we can get an estimate of the average path length of the ERG ensemble build by using the degree sequence as the observable.

We can retrieve equation (3.5) from (3.6) by considering the observables as the quantities we want to estimate. As stated before, by doing this we can obtain a close form to calculate analytically or numerically the parameters $\beta_i$. In fact following equation (3.1) we obtain

$$\langle x_i\rangle = \frac{1}{Z}\sum_{G\in\mathcal{G}} x_i(G)e^{\sum_{j=1}^{r}\beta_j x_i(G)}. \tag{3.7}$$

We have now that

$$\langle x_i \rangle = \frac{1}{Z} \sum_{G \in \mathcal{G}} x_i(G) e^{\sum_{j=1}^{r} \beta_j x_i(G)} = \frac{1}{Z} \frac{\partial}{\partial \beta_i} \sum_{G \in \mathcal{G}} e^{\sum_{j=1}^{r} \beta_j x_i(G)} = \frac{1}{Z} \frac{\partial Z}{\partial \beta_i} = \frac{\partial F}{\partial \beta_i}, \tag{3.8}$$

where $F := \ln Z$ is called the *free-energy* of the model. Moreover, since it also holds the following

$$\langle {x_i}^2 \rangle = \frac{1}{Z} \sum_{G \in \mathcal{G}} x_i(G)^2 e^{\sum_{j=1}^{r} \beta_j x_i(G)} = \frac{1}{Z} \frac{\partial^2 Z}{\partial {\beta_i}^2}, \tag{3.9}$$

we can also retrieve a close form for the variance of each observable:

$$\langle {x_i}^2 \rangle - (\langle x_i \rangle)^2 = \frac{1}{Z} \frac{\partial^2 Z}{\partial {\beta_i}^2} - \Big( \frac{1}{Z} \frac{\partial Z}{\partial \beta_i} \Big)^2 = \frac{\partial}{\partial \beta_i} \Big( \frac{1}{Z} \frac{\partial Z}{\partial \beta_i} \Big) = \frac{\partial^2 F}{\partial {\beta_i}^2} = \frac{\partial \langle x_i \rangle}{\partial \beta_i}. \tag{3.10}$$

In statistical physics this last expression takes the name of *fluctuation-response relation*.

### 3.2.2 ERG distribution from maximum entropy

We will now briefly show the rationale behind the form of the ERG model given by equation (3.1) from an Information Theory perspective. Consider the general problem of having a graph $G$ with some measurable properties (the observables) $x_1^*, x_2^*, \ldots, x_r^*$. Let $\mathcal{G}$ be the set of all possible network realizations (the ensemble). We want to define a probability measure $P(\cdot)$ over $\mathcal{G}$ so that the expectation of each of the observable computed over all the possible network realizations should be equal to the observed value $x_i(G)$. The probability measure should be the *best* choice we can make, *i.e.,* the one that uses the minimum assumptions while also satisfying each constraint that we can possibly have in our framework. According to the maximum entropy principle of information theory [29], the *best* choice of $P(\cdot)$ is given by the one that maximizes the Shannon/Gibbs entropy,

$$S = -\sum_{G \in \mathcal{G}} P(G) \ln P(G), \tag{3.11}$$

subjected to the constraint that $\sum_{G \in \mathcal{G}} P(G) = 1$ and the ones arising from (3.5). This is due to the fact that (3.11) measures precisely the opposite of the information used, that means that the bigger is the value of $S$ the least amount of assumptions are used.

Since we are now in the situation of having a constrained maximization problem, we can use the Lagrange multipliers to find a solution. We introduce the multiplier $\lambda$, for the first

constraint arising from the fact that $P()$ should be a probability measure, and the multipliers $\beta_1, \beta_2, \ldots, \beta_r$, for the constraints arising from (3.5), then the maximum value for $S$ is achieved when $P()$ satisfies:

$$\frac{\partial}{\partial P(G)}\Big[S - \lambda\Big(1 - \sum_{G\in\mathcal{G}} P(G)\Big) - \sum_{i=1}^{r} \beta_i\Big(x_i^* - \sum_{G\in\mathcal{G}} x_i(G)P(G)\Big)\Big] = 0 \quad \forall G \in \mathcal{G}. \tag{3.12}$$

This gives

$$-\ln P(G) - 1 + \lambda + \sum_{i=1}^{r} \beta_i x_i(G) = 0, \tag{3.13}$$

which implies the form given by equation (3.1) for $P(G)$,

$$P(G) = \exp\Big[\lambda - 1 + \sum_{i=1}^{r} \beta_i x_i(G)\Big] = \frac{e^{H(G)}}{Z}, \tag{3.14}$$

with $H(G) = \sum_{i=1}^{r} \beta_i x_i(G)$ being what we have defined as the Hamiltonian and $Z = e^{1-\lambda}$ being the normalizing partition function.

## 3.3 Monte-Carlo methods for ERG

We have seen that if we have a closed-form expression for the partition function $Z$ we can then compute the parameters of the model using (3.8), and also sampling each graph directly using (3.1) since it depends only on the graph observables and on the partition function $Z$. In practice, however, there are very few model specifications that lead to a tractable form for the partition function, and thus the parameters estimation cannot be done analytically. On the other hand, sampling from the model does not require the knowledge of the partition function, since it can be done via Monte-Carlo simulations, as we will describe in the following.

### 3.3.1 Metropolis-Hastings for Exponential Random Graph

First we recall briefly how the Metropolis-Hastings (MH) algorithm works, since we will rely on it. MH is one of the most popular Markov-Chain Monte Carlo (MCMC) methods.

Let $X$ be a state space and assume we want to generate samples from a known probability distribution $\mathcal{P}(x)$ for each $x \in X$. We want to build a Markov process $M_t$ that moves in $X$ with a unique stationary distribution $\pi(x) = \mathcal{P}(x)$. A Markov process is uniquely defined

by its transition probabilities $P(x \to x')$, i.e., the probability of going from state $x$ to state $x'$. In the following we will consider, unless stated otherwise, only ergodic Markov processes, that means aperiodic and positive recurrent processes.

For these processes, a sufficient but not necessary condition to guarantee the existence of the stationary distribution is the so called *detailed balance equation*:

$$\pi(x')P(x \to x') = \pi(x')P(x' \to x), \tag{3.15}$$

which means that the probability of going from state $x$ to state $x'$ multiplied by the probability of being in state $x$ is equal to the probability of going from state $x'$ to state $x$ multiplied by the probability of being in state $x'$.

As mentioned earlier, the Metropolis-Hastings algorithm aims to build a process whose unique stationary distribution is $\pi(x) = \mathcal{P}(x)$, and to do so the algorithm starts from the *Detailed balance equation* (3.15)

$$\mathcal{P}(x')P(x \to x') = \mathcal{P}(x')P(x' \to x), \tag{3.16}$$

which can be rewritten as

$$\frac{P(x \to x')}{P(x' \to x)} = \frac{\mathcal{P}(x')}{\mathcal{P}(x')}. \tag{3.17}$$

Now the Metropolis-Hasting algorithm separates the transition probability into a proposal $T(x \to x')$ and an acceptance probability $A(x \to x')$: the proposal distribution $T(x \to x')$ is the conditional probability of going from state $x$ to state $x'$, the acceptance distribution $A(x \to x')$ is the probability of accepting state $x'$ as the new state that the chain is visiting.

The transition probability can thus be rewritten as

$$P(x \to x') = T(x \to x')A(x \to x'), \tag{3.18}$$

and now by plugging (3.18) into (3.17) we obtain

$$\frac{A(x' \to x)}{A(x \to x')} = \frac{\mathcal{P}(x')T(x \to x')}{\mathcal{P}(x')T(x' \to x)}. \tag{3.19}$$

The next step is to find an acceptance probability $A()$ that satisfies equation (3.19). The Metropo-

lis choice to solve this problem is given by

$$A(x' \to x) = \min\left\{1, \frac{\mathcal{P}(x')T(x \to x')}{\mathcal{P}(x)T(x' \to x)}\right\}. \tag{3.20}$$

For the Metropolis acceptance ratio either $A(x \to x') = 1$ or $A(x' \to x) = 1$, and then the condition (3.19) is satisfied.

The Metropolis-Hastings algorithm can be easily adapted for the Exponential Random Graphs models, in particular the form of the acceptance ratio makes so that it is possible to run the algorithm without knowledge of the partition function $Z$. In fact the Metropolis-Hastings algorithm for an ERG can be written in the following way:

**Algorithm 1** Metropolis-Hastings Algorithm for Exponential Random Graphs

Start from $G^0 = (V^0, E^0) \in \mathcal{G}$
**for** $k = 1, \dots, K$ **do**
  Generate a random edge $(i, j)$
  **if** $(i, j) \in E$ **then**
    Remove the edge: $E^k = E^{k-1} \setminus (i, j)$
    accept $G^k$ with probability $A = \min\left\{1, P(G^k)/P(G^{k-1})\right\}$
  **else**
    Add edge: $E^k = E^{k-1} \cup (i, j)$
    accept $G^k$ with probability $A = \min\left\{1, P(G^k)/P(G^{k-1})\right\}$
  **end if**
**end for**

Notice that since $A = \min\left\{1, P(G^k)/P(G^{k-1})\right\}$, we can rewrite this acceptance probability as

$$A = \min\left\{1, \frac{e^{H(G^k)}}{e^{H(G^{k-1})}} \cdot \frac{Z}{Z}\right\}, \tag{3.21}$$

and thus we can simplify by removing completely the contribution of $Z$, obtaining

$$A = \min\left\{1, e^{H(G^k) - H(G^{k-1})}\right\}, \tag{3.22}$$

which means that the acceptance probability is equal to one if $H(G^k) - H(G^{k-1}) > 0$ and equal to $H(G^k) - H(G^{k-1})$ otherwise. By using Algorithm 1 with parameters $\beta$ that sat-

isfy equations (3.8), after convergence is reached we are guarantee to sample from an ensemble whose considered observables, *i.e.,* the ones that we include in the Hamiltonian, have on average the same values as the ones of the observed graph.

If we do not have a closed-form expression of the partition function to calculate the "true" values of the parameters, we need to approximate them. We must take into account that, especially when we include in the Hamiltonian terms that go beyond the dyadic relationships of the nodes (*e.g.* when we include triangle or $k-$stars), this can lead to odd behaviours of the simulated Exponential Random Graph models, ending with degenerate graphs *i.e.,* nearly full or nearly empty graphs.

### 3.3.2 Phase transitions in ERGMs

Phase transitions are another phenomenon that characterize Exponential Random Graphs models and have also a very specific interpretation from a physical point of view [24]: with some model specifications, it has been observed that very similar parameters' configurations could lead to completely different topologies, dividing the space of possible outcomes of the given model specification into two "phases", with each phase consisting of a sub-enesemble of graphs that share a macro-property, for example is fairly common to observe a phase of very sparse graphs and a phase of nearly complete graphs both arising from the same specified model with similar parameters. This behaviour have another very concerning implication, since a model specification exhibiting a phase transition could make some outcomes impossible to reach, effectively meaning that in some situations the desired ensemble could not be build using some ERG specifications. In the physical systems phase transitions give rise to interesting phenomena such as ferromagnetism or superconductivity.

A possible theoretical explanation of phase transition in Exponential Random Graphs when dealing with dense graphs can be retrieved from the work of Chatterjee and Diaconis [30].

Since, as we will see in chapter 5, we will work with model specifications for which we do not have a closed-form expression to compute the parameters, and in fact we will use Monte-Carlo methods to estimate the values of these parameters, we need to be aware of both the possibility of degeneracy and analyzing phase transitions if arising.

# 4

# Analysis of available grids

Before going through the modeling phase we analyze here the grids' datasets that we used as a reference to develop our models. The grids were collected and described in [31] and are available in a *MATPOWER testcase* format [32] in this Github repository power grid optimal power flow library. The analyses contained in this chapter have been performed using the programming language Python 3 [33].

## 4.1 Grid data

The parsing of these files was done using this library created by Leon Lan. For each grid we obtain a Pandas dataframe [34] objects for the buses' data, the branch data and the generator data. From these dataframes we then build up a simple, undirected, unweighted graph object with the bus type as a node attribute, and the associated adjacency matrix. The bus type were inferred as follows: the generators are retrieved directly from the generator list available in the MATPOWER file, the nodes regarded as interconnections are the ones with 0 power generation and 0 power demand, and the other nodes are labeled as loads. As stated before, we choose to neglect some of the electrical aspects of the grids, such as line capacities and generation/demand allocation, during the development of our models since our main objective is to generate realistic synthetic topologies, however our procedure is able to retrieve also all the electrical characteristics from the MATPOWER file, thus enabling future works on these aspects.

| Name | $N$ | $\|E\|$ | $\langle k_{\text{gen}} \rangle$ | $\langle k_{\text{load}} \rangle$ | $\langle k_{\text{int}} \rangle$ |
|---|---|---|---|---|---|
| 118_ieee.m | 118 | 186 | 3.56 | 2.50 | 3.25 |
| 1354_pegase.m | 1354 | 1991 | 2.58 | 1.08 | 2.53 |
| 162_ieee_dtc.m | 162 | 284 | 1.67 | 3.86 | 2.71 |
| 179_goc.m | 179 | 263 | 1.00 | 2.45 | 3.65 |
| 1888_rte.m | 1888 | 2531 | 0.82 | 2.76 | 1.76 |
| 1951_rte.m | 1951 | 2596 | 0.92 | 2.78 | 4.12 |
| 2000_goc.m | 2000 | 3639 | 1.15 | 3.08 | 2.78 |
| 200_activ.m | 200 | 245 | 1.00 | 2.91 | 3.25 |
| 2312_goc.m | 2312 | 3013 | 2.01 | 2.51 | 2.71 |
| 2383wp_k.m | 2383 | 2896 | 3.01 | 2.33 | 0.00 |
| 240_pserc.m | 240 | 448 | 1.00 | 3.44 | 0.00 |
| 2736sp_k.m | 2736 | 3504 | 3.44 | 2.45 | 3.00 |
| 2737sop_k.m | 2737 | 3506 | 3.56 | 2.45 | 3.50 |
| 2742_goc.m | 2742 | 4673 | 3.67 | 2.91 | 2.07 |
| 2746wop_k.m | 2746 | 3514 | 3.20 | 2.45 | 2.80 |
| 2746wp_k.m | 2746 | 3514 | 3.16 | 2.45 | 0.00 |
| 2848_rte.m | 2848 | 3776 | 8.14 | 1.18 | 0.04 |
| 2853_sdet.m | 2853 | 3921 | 1.90 | 2.85 | 2.87 |
| 2868_rte.m | 2868 | 3808 | 9.22 | 0.81 | 0.00 |
| 2869_pegase.m | 2869 | 4582 | 2.70 | 2.62 | 2.79 |
| 300_ieee.m | 300 | 411 | 1.96 | 3.06 | 2.15 |
| 3012wp_k.m | 3012 | 3572 | 2.96 | 2.29 | 1.22 |
| 3022_goc.m | 3022 | 4135 | 1.68 | 2.76 | 2.85 |
| 30_as.m | 30 | 41 | 2.00 | 2.77 | 4.50 |
| 30_ieee.m | 30 | 41 | 2.00 | 2.77 | 4.50 |
| 3120sp_k.m | 3120 | 3693 | 2.92 | 2.30 | 1.22 |
| 3375wp_k.m | 3374 | 4161 | 3.54 | 1.81 | 8.00 |
| 3970_goc.m | 3970 | 6641 | 3.34 | 2.86 | 2.54 |
| 39_epri.m | 39 | 46 | 1.10 | 2.79 | 0.00 |
| 4020_goc.m | 4020 | 6988 | 3.74 | 3.02 | 2.49 |
| 4601_goc.m | 4601 | 7199 | 3.33 | 2.72 | 3.24 |
| 4619_goc.m | 4619 | 8150 | 3.78 | 3.19 | 2.50 |
| 4661_sdet.m | 4661 | 5997 | 2.45 | 2.43 | 2.78 |
| 4837_goc.m | 4837 | 7765 | 3.22 | 2.72 | 2.54 |
| 4917_goc.m | 4917 | 6726 | 1.85 | 2.71 | 2.85 |
| 500_goc.m | 500 | 733 | 1.38 | 3.22 | 2.10 |
| 57_ieee.m | 57 | 80 | 3.86 | 2.62 | 2.00 |
| 588_sdet.m | 588 | 686 | 2.54 | 2.18 | 2.87 |
| 73_ieee_rts.m | 73 | 120 | 2.88 | 3.11 | 2.00 |
| 793_goc.m | 793 | 913 | 2.61 | 2.15 | 2.48 |
| 89_pegase.m | 89 | 210 | 3.17 | 7.15 | 3.14 |

**Table 4.1:** Available grids after parsing

| Name | $N$ | $\lvert E\rvert$ | $\langle k_{\text{gen}}\rangle$ | $\langle k_{\text{load}}\rangle$ | $\langle k_{\text{int}}\rangle$ |
|---|---|---|---|---|---|
| 10000_goc.m | 10000 | 13193 | 1.33 | 2.69 | 3.57 |
| 10480_goc.m | 10480 | 18559 | 3.74 | 3.05 | 3.02 |
| 13659_pegase.m | 13659 | 20467 | 1.02 | 1.53 | 3.64 |
| 19402_goc.m | 19402 | 34704 | 3.75 | 2.99 | 3.46 |
| 24464_goc.m | 24464 | 37816 | 3.33 | 2.90 | 2.80 |
| 30000_goc.m | 30000 | 35393 | 1.54 | 2.36 | 3.75 |
| 6468_rte.m | 6468 | 9000 | 1.83 | 2.60 | 3.27 |
| 6470_rte.m | 6470 | 9005 | 1.81 | 2.59 | 4.82 |
| 6495_rte.m | 6495 | 9019 | 1.78 | 2.56 | 5.95 |
| 6515_rte.m | 6515 | 9037 | 1.77 | 2.57 | 5.90 |
| 8387_pegase.m | 8387 | 14561 | 3.26 | 3.01 | 3.74 |
| 9241_pegase.m | 9241 | 16049 | 3.21 | 1.92 | 3.12 |
| 9591_goc.m | 9591 | 15915 | 3.74 | 2.92 | 2.43 |

**Table 4.2:** Available grids after parsing

At the end of this procedure we obtained 54 different grids of various sizes, ranging from small (14 buses) to large (30000 buses). We report here in Table 4.1 and Table 4.2 the list of all these grids with name, number of buses ($N$), number of lines ($|E|$), and average node degree per bus type ($\langle k_{\text{gen}}\rangle$, $\langle k_{\text{load}}\rangle$, $\langle k_{\text{int}}\rangle$).

Because of the sensitive nature of the data about real power systems, even the publicly available grids are often only partial representations of real grids: in order to avoid any disclosure of important information, they might been build by sampling a bigger grid in order to obtain a smaller pseudo-realistic one or they might even be partially synthetic themselves. This means that is crucial to understand the origin of the grids we ended up with after the procedure described above before going on, so that we can assess the quality of the data we are working with and understanding if it fits the scope of our research. Thus we briefly describe now the obtained grids and their original source:

- the IEEE testcases [35] represent portions of the American Electric Power System (in the Midwestern US) extracted in the period 1960-1965. These networks are small (14 buses) to medium (300 buses) size.

- the RTE and PEGASE testcases [36] represent accurately the size and complexity of portions of the European grids (in particular the high-voltage and super high-voltage French grids are used for the RTE grids whereas mid-European grid are used for the PEGASE). Albeit accurate from an electrical perspective, these cases are fictitious, often obtained through sampling of the real grids, thus we must be careful when using them as a reference for the topological properties.

- the Polish test cases represent the high-voltage transmission lines during winter peak conditions ("wp") in 2007-2008, winter off-peak conditions ("wop") in 2007-2008, summer peak conditions ("sp") and summer off-peak conditions ("sop") in 2004. They are all reduced grids, representing only partially the real network. The topologies might not be entirely reliable since multiple nodes were aggregated and synthetic nodes were added.

- The "goc" (*Grid Optimization Competition*) and "sdet" (*Sustainable Data Evolution Technology*) are completely synthetic, and most of them are generated through the procedure described in [3] (most of the "goc" grids) or using the SDET procedure described in [21] (all the "sdet" grids, some of the "goc" grids). When using this networks we must be aware of their synthetic nature, thus we should avoid using them to validate our methods.

## 4.2 Descriptive analysis

Because of the ERG structure, we have decided to analyze some specific topological aspects that can be relevant for our models, namely triangles and $k-$triangles and the differences among nodes belonging to different bus types. For the other properties like average shortest path length, algebraic connectivity or average global degree we have found results consistent with the literature, thus we omit here the detailed results and we refer to the articles that describe in depth these characteristics [12, 7, 19].

### 4.2.1 Triangles and K-triangles

One of the main topological properties of the power grids is that the global clustering coefficient have been observed to be high when compared to random graphs of similar size [11]. Since the global clustering coefficient is computed as in (2.7), an higher value of this property means that the graph presents a number of triangles that is higher than the other sparse graphs. In Fig. 4.1 we can see the relation between the number of edges ($|E|$) and the number of triangles ($|T_1|$) for the 54 networks in Table 4.1.

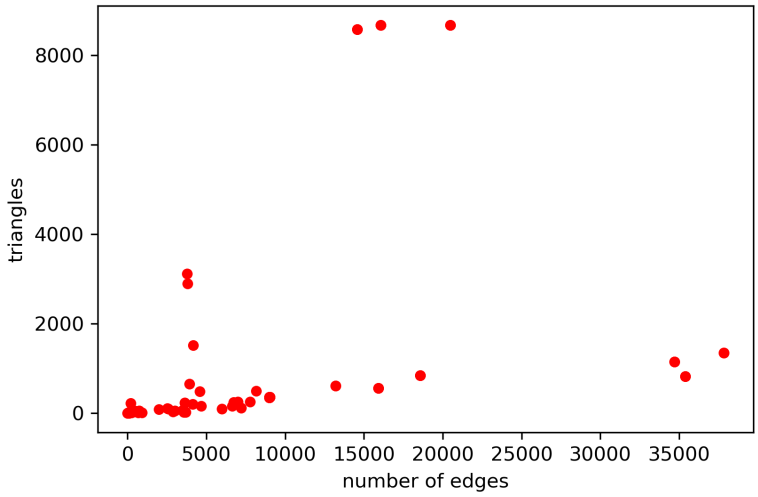


**Figure 4.1:** Relation between number of edges and number of triangles in Table 4.1

It is worth highlighting two distinct behaviours: the majority of the grids have a number of triangles that is much smaller than the number of edges, however some grids have a much higher number of triangles compared to the others ($\frac{|E|}{2} < |T_1|$). We further investigate this phenomenon by looking at another quantity that have been used also in the ERG literature [37]: the $k-$triangles. A $k-$triangle is defined as $k$ different triangles that share the same edge. Examples of this structure are given in Fig. 4.2

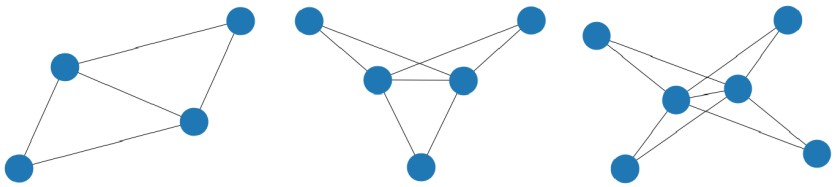

**Figure 4.2:** A $2-$triangle, a $3-$triangle and a $4-$triangle

From a topological point of view analyzing the $k-$triangles capture the nestedness of the triangles in the graph, giving insights on the global structure. In particular, we look into the relation between the number of edges and the number of $2-$triangles ($|T_2|$). This choice was made because of it is the most simple of the $k-$triangular structures and furthermore because the number of $2-$triangles determines the number of $k-$triangles with $k > 2$. It is also important to notice that is possible to have $|T_2| > |T_1|$ if there are multiple $k-$triangles with high $k$,

since any $k-$triangle is composed of $\binom{k}{2}$ $2-$triangles (*e.g.* consider the $4-$triangle in Fig. 4.2, in that structure we have $|T_2| = \binom{4}{2} = 6$). The scatterplot in Fig. 4.3 shows the relation between number of edges and $2-$triangles in the grids.

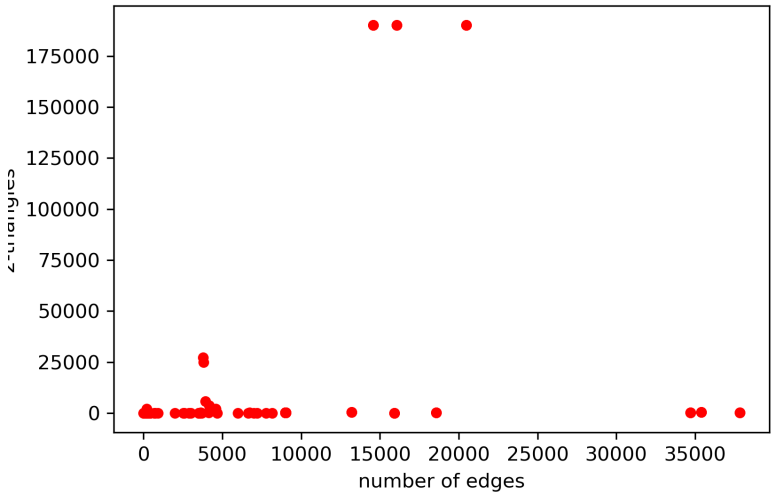


**Figure 4.3:** Relation between number of edges and number of $2-$triangles

The last scatterplot furthermore confirms the existence of two regimes in the topology of the grids. Motivated by this fact, we introduce a distinction between the networks: we say that a graph is *supertriangular* if $|T_2| > |T_1|$ and *normotriangular* otherwise. We have found 12 supertriangular networks among the 54, and of these 6 are from PEGASE test cases, that are nested within each other (effectively meaning that they corresponds to portions of the same aggregated grid/ they have synthetic buses and lines added with the same procedure). The reason behind the existence of these two behaviour should be further investigated. It is probable, albeit not provable without more data, that the presence of multiple $k-$triangles in a grid is due to the necessity to satisfy the $N-1$ robustness conditions, that translates to the requirement of $2-$connectivity from a topological point of view. In fact a $k-$triangular structure allows for better redundancy (in fact there will still exist a path between any two node of these structures if any single edge is removed). However since this could also be symptomatic of a myopic procedure to synthesize grids or to modify portions of existing grids in order to avoid restrictions, we choose to work exclusively with the networks that we have said to be normotriangular, that represent also the majority of the available grids. We now regenerate Fig. 4.1 and Fig. 4.3 focusing only on normotriangular networks.

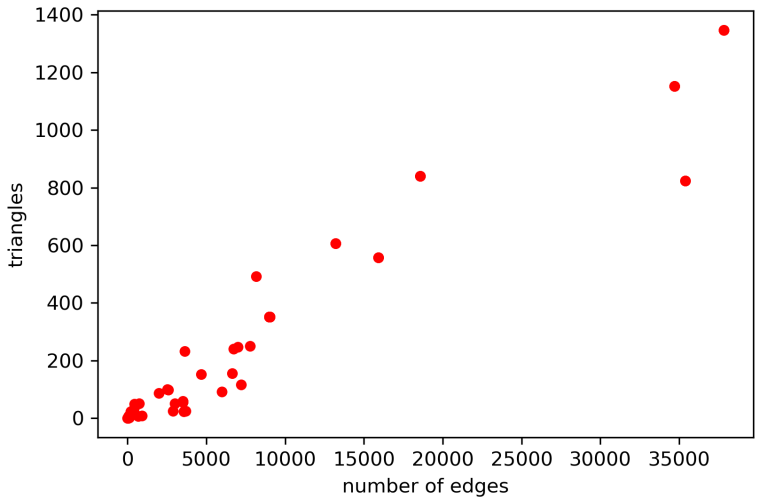


**Figure 4.4:** Relation between number of edges and number of triangles for the normotriangular networks

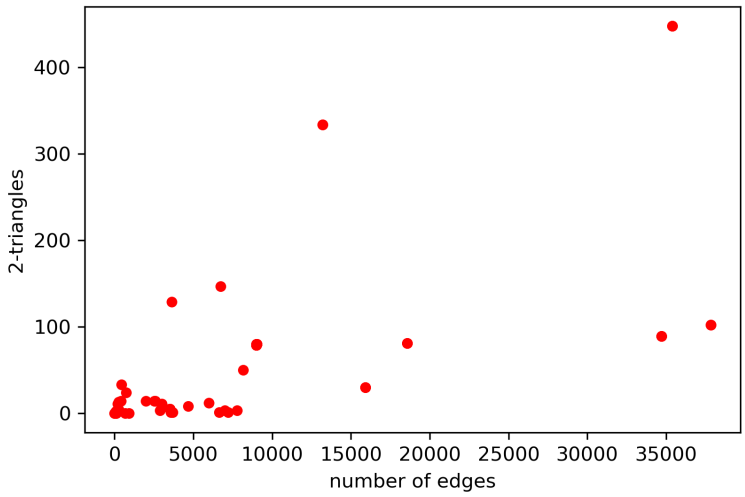


**Figure 4.5:** Relation between number of edges and number of $2-$triangles for the normotriangular networks

As we can see in Fig. 4.4, without the supertriangular graphs there is a clear relation between number of edges and number of triangles, and this is also true up to an extent for the $2-$triangles, see Fig. 4.5. It is also worth mentioning that for almost every grid we have $|T_2| < 100$, meaning that regardless of the network size we have a number of $2-$triangles that is small

and almost constant (it is also important to notice that this implies also that we should expect the number of $k-$triangles with $k > 2$ to be even smaller, negligible for most of the grids).

### 4.2.2 Bus type percentages

It has already been observed in [12] that the average degree is different for each bus-type, thus when modeling the grids' topologies we should take into account this aspect over just considering the global average degree. To better understand the differences between generators, loads, and interconnections, we compute the percentages of each bus type to all the nodes with respect to the network size among the grids selected. By doing so we want to highlight the importance of a correct bus type assignment during the modeling, and also it can be useful to further improve our understanding of the networks at our disposal, as done before with the triangular counts.

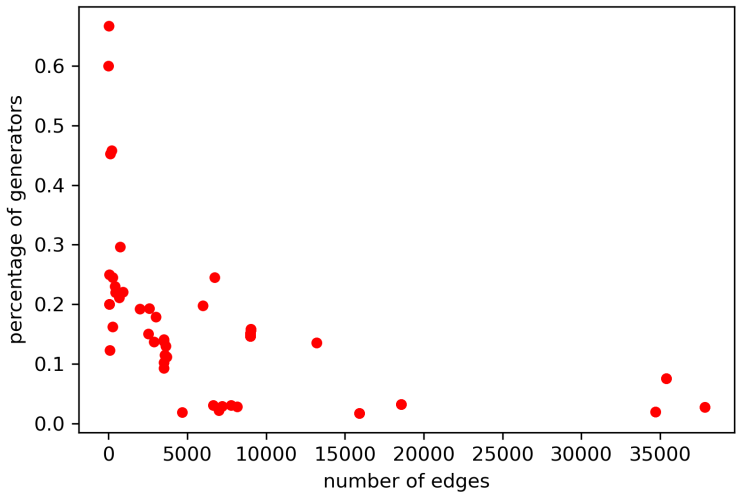


**Figure 4.6:** Fraction of generators in the graph to number of edges

In Fig. 4.6 we can see that the bigger the network, the smaller the percentage of generators. This could be due either to the fact that the smaller grids are just portions of the bigger ones and thus to obtain a relevant test case the generators tend to be included more than the other nodes, or because in many bigger grids multiple generators can be aggregated into one node. On the converse, as we can see in Fig. 4.7 the number of loads tends to increase with the network size. This could be due to the same reasons as for the behaviour of the percentage of generators,

but with a reversed effect: in fact for the smaller grids loads can be neglected or aggregated, whereas bigger grids have multiple loads since they should represent more complex geographical features.

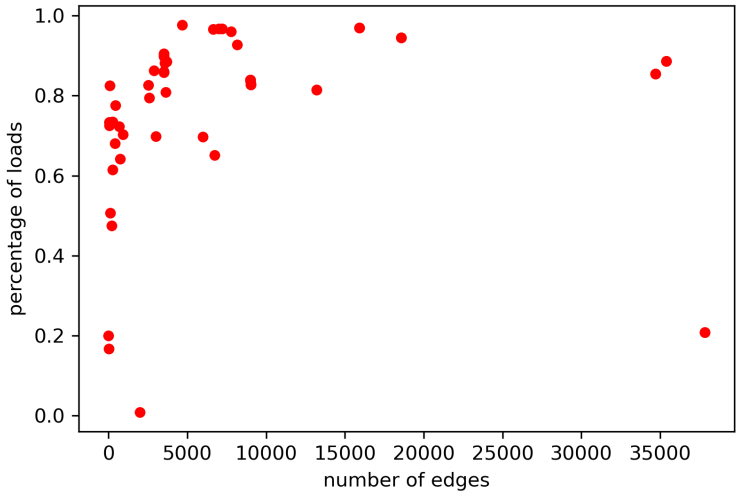


**Figure 4.7:** Percentage of loads in the graph to number of edges

The percentage of interconnections, as we can see in Fig. 4.8 is almost constant regardless of the size, with two exceptions represented by the `1354 PEGASE` and the `30000 PEGASE`, where the percentage of interconnections is close to 90%. The reason behind this fact is unclear. It is possible that this is due to poor labeling of the nodes or because of a particular form of aggregation of generators and loads.

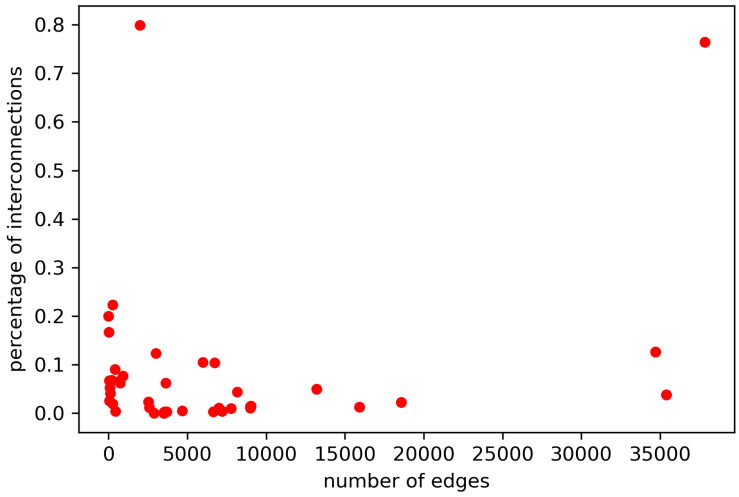


**Figure 4.8:** Percentage of interconnections in the graph to number of edges

## 4.3 Final considerations

The results of our analysis showed a seemingly coherent behaviour of the available grids with respect to the considered properties, when we restrict ourselves to those networks for which it holds $|T_2| < |T_1|$. Moreover, we highlight the oddity of the grids belonging to the PEGASE type, which we recall to be a collection of nested and fictitious grids. For this reason we choose, when possible, to avoid using them as a reference for our models.

# 5

# ERGMs for power systems

This chapter will include the main results of our work, both from a power system modeling and from ERG theory point of view.

- In Section 5.1 we describe the first model specification that we propose, including the proof of one theorem relative to an ERG class which our specification belongs to and the results that we obtained.

- In Section 5.2 we show our second and final model that arise from solving the main issues of the specification proposed before. This section is organized as follows: first we specify the form of the Hamiltonian, explaining the reasons behind the new considered observables. Then we move into the problem of estimating the parameters of this model. For this reason, we first present the methods already present in the literature, then we highlight their limits within our problem. We then prove the main theoretical result of this model, which is a MCMC inspired method to estimate the parameters of an ERG ensemble of connected graphs. Finally, we show the results of this last model with the parameters estimated with our method.

- In Section 5.3 we compare the results obtained with our approach to the real grids of reference, and we comment the advantages and the limits of our models both from a theoretical and from a computational point of view.

## 5.1 Model I: Edge-Types Model

One of the crucial aspects of a power network is the so-called *bus-type assignment*. As mentioned in Chapter 2, when modeling each bus as a node of a network it must be taken into account that there are three type of bus, usually referred to as Generator ($P$), Load ($L$) and Interconnection ($I$), each with a different role in the system and thus each exhibiting different nodal properties.

In the literature many proposals were formulated to solve the bus-type assignment problem, for example in [14] the authors introduce an algorithm to assign the proper bus-type at each node after generating the whole topology based on a measure called "Bus-Type Entropy" first introduced in [13].

Within our framework, however, it seems better to incorporate the information on the bus-types directly into the model. Given a a simple, undirected, unweighted graph $G = (V, E)$ representing a power grid, we can consider a partition of $E$ based on the bus-type of the nodes connected by each edge. With the 3 different possible bus-types described above, this partition leads to 6 different type of edges that we name $E_{PP}, E_{PL}, E_{PI}, E_{LL}, E_{LI}, E_{II}$, where $E_{ab}$ indicates that the edge connects a node of type $a$ to one of type $b$.

We propose now an ERG model with the following Hamiltonian specification:

$$\mathcal{H} = \beta_{PP}|E_{PP}| + \beta_{PL}|E_{PL}| + \beta_{PI}|E_{PI}| + \beta_{LL}|E_{LL}| + \beta_{LI}|E_{LI}| + \beta_{II}|E_{II}|. \quad (5.1)$$

This specification not only solves the bus type assignment by including as observables the number of edges for each type, but has also the huge advantage of leading to a closed-form expression for the partition function, as we will prove in the following section.

### 5.1.1 Theoretical results for Model I

Recall now that if we consider a ERG model with just the number of edges as an observable it is possible to compute the exact partition function and the resulting model will be equivalent to an Erdös-Renyi model [24]. The following theorem that we prove generalizes this idea to a wider class of models, that includes also our specification.

**Theorem 1.** *Consider a simple, undirected, unweighted graph $G = (V, E)$, and let $A = \big(A_{ij}\big)$ be the symmetric adjacency matrix associated to G.Let $E_1, E_2, E_3, \ldots, E_K$ be a partition of $E$ and let $A_1, A_2, A_3, \ldots, A_K$ be a block decomposition of $A$ such that $A_m = \big(A_{ij}\big)\forall(i, j) \in E_m$.*

*Then for an Exponential Random Graph Model with Hamiltonian $H$, defined by*

$$H = \sum_{i}^{K} \beta_i |E_i| \tag{5.2}$$

*where $\beta_i$ are real parameters and $|E_i|$ indicates the number of nonzero entries in $A_i$, the partition function $Z$ associated to $H$ takes the following form:*

$$Z = \prod_{i}^{K} (1 + e^{\beta_i})^{M(E_i)}, \tag{5.3}$$

*where $M(E_i)$ indicates the value that $E_i$ would take if $G$ were a complete graph.*

*Proof.* Using the Hamiltonian (5.2), the partition function of this model is

$$Z = \sum_{G \in \mathcal{G}} e^{\sum_i^K \beta_i |E_i(G)|}. \tag{5.4}$$

The key observation here is that when summing over all the possible graphs, each block $A_i$ can be considered as an independent matrix and $|E_i(G)|$ represents the number of edges in the portion of graph associated with $A_i$. We can thus rewrite (5.4) as

$$Z = \sum_{G \in \mathcal{G}} \prod_{i}^{K} \prod_{A_{ij} \in A_i} e^{\beta_i A_{ij}}, \tag{5.5}$$

but now since $A_{ij}$ can take only values in $\{0, 1\}$ and in each block $A_i$ we have $M(E_i)$ possible entries, we obtain

$$Z = \prod_{i}^{K} \prod_{A_{ij} \in A_i} (1 + e^{\beta_i}) = \prod_{i}^{K} (1 + e^{\beta_i})^{M(E_i)}. \tag{5.6}$$

□

With this formulation of the partition function we can then easily compute the *free-energy*

$$F = \log Z = \log \prod_{i}^{K} (1 + e^{\beta_i})^{M(E_i)} = \sum_{i}^{K} M(E_i) \log (1 + e^{\beta_i}), \tag{5.7}$$

and recalling (3.8) we have for each $i$ it also holds

$$\langle |E_i| \rangle = \frac{\partial F}{\partial \beta_i} = \hat{A}_i \frac{e^{\beta_i}}{1 + e^{\beta_i}}. \tag{5.8}$$

And as said in Chapter 3, this formulation can be used to desired find the values of the parameters by imposing $\langle |E_i| \rangle = |E_i|$.

It is easy to see now that in our case the following Hamiltonian specification satisfies the assumptions of Theorem 1

$$H = \beta_{PP}|E_{PP}| + \beta_{PL}|E_{PL}| + \beta_{PI}|E_{PI}| + \beta_{LL}|E_{LL}| + \beta_{LI}|E_{LI}| + \beta_{II}|E_{II}|. \tag{5.9}$$

This specification assures that the bus type assignment will be on average the same as the one of the true network when the parameters $\beta$ are obtained with the method described above.

### 5.1.2 Computational results for Model I

As stated in Section 3.3 in order to simulate from an ERG with a given parameter configuration we can use the Metropolis-Hastings algorithm.

Here as an example we report the results of the simulations from model (5.9) with the "300 ieee" as an input network. As stated in Chapter 4, the "300 ieee" is a test case that represents a portion of the American Electric Power System (in the Midwestern US) extracted in the period 1960-1965. It has 300 nodes, 411 edges, 69 generators, 204 loads and 27 interconnections. For the edge type counts, using the notation introduced for (5.9), this grids exhibits the following values: $E_{PP} = 8, E_{PL} = 110, E_{PI} = 9, E_{LL} = 240, E_{LI} = 35, E_{II} = 7$.

We use Metropolis-Hastings algorithm with the set of parameters retrieved from applying Theorem 1 and ended up with $\approx 160000$ synthetic graphs collected after reaching the steady state distribution, *i.e.,* when the average values of the considered observables of the samples is close to those used to estimate the parameters. Of these samples we make a further strict selection to obtain "weakly-correlated samples", according to the rule that two matrices associated to different sampled graphs should have an Hamming distance of at least $2N$. After this selection we obtained 2458 weakly-correlated samples corresponding to networks with a bus-type assignment similar to the one of the `300 ieee`. The whole procedure took 40 minutes.

In the following we report the histograms referring to the distribution of some statistics calculated on the 2458 samples obtained with the model.

In Figs. 5.1 to 5.3 we see the distributions of the degrees of the three bus-types, the red line

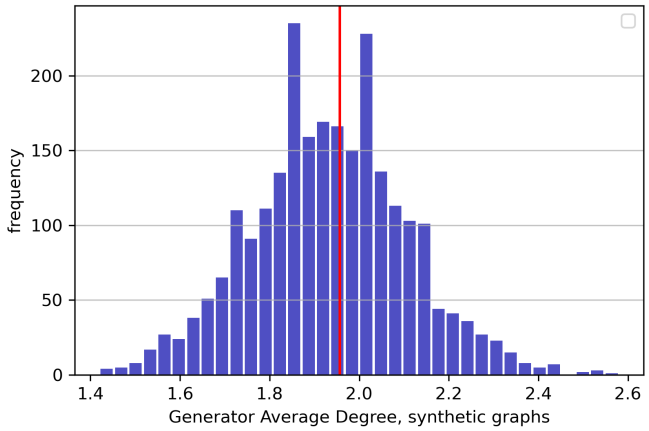


**Figure 5.1:** Generator average degree

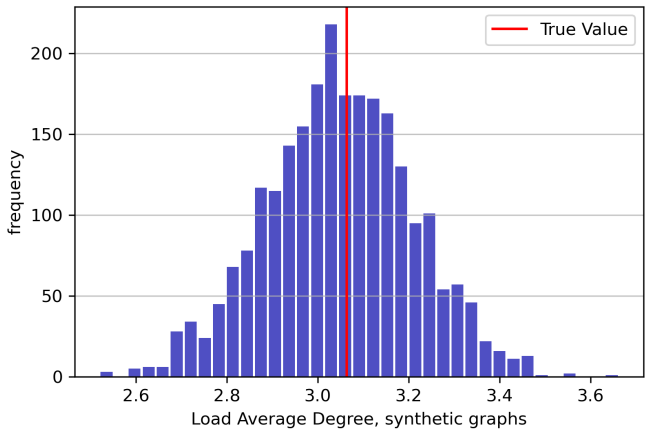


**Figure 5.2:** Load average degree

representing the true value calculated on the real grid, and the green one representing the average of the same value calculated on the generated graphs. We can see that on average the samples have the same mean degrees as the real network. This is a promising result considering the simplicity of the model.

However, the histogram in Fig. 5.4 highlights one of the main problems that arise from such a simple model specification.

In Fig. 5.4 we see that the number of triangles (average value of the generated grids highlighted in green), which in turn determines the global clustering coefficient, is much lower in

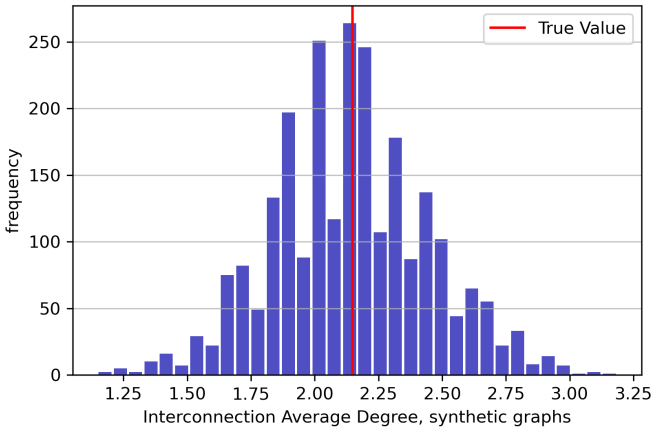


**Figure 5.3:** Interconnection average degree

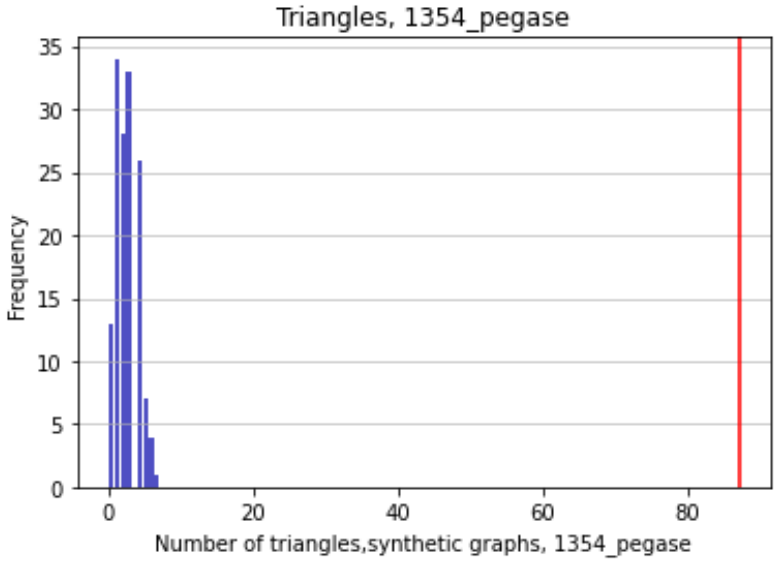


**Figure 5.4:** Number of triangles

the synthetic graphs than in the real one (the value highlighted in red).

Another fundamental aspect that we should observe is that this model does not guarantee by any means the connectivity of the generated graphs. In fact in Fig. 5.5 we show how the number of connected components of a graph generated by this method varies between 10 to 46 for the "300 ieee". Further results obtained with other real grids suggest that this method generates on average graphs with $\approx \frac{N}{10}$ connected components.

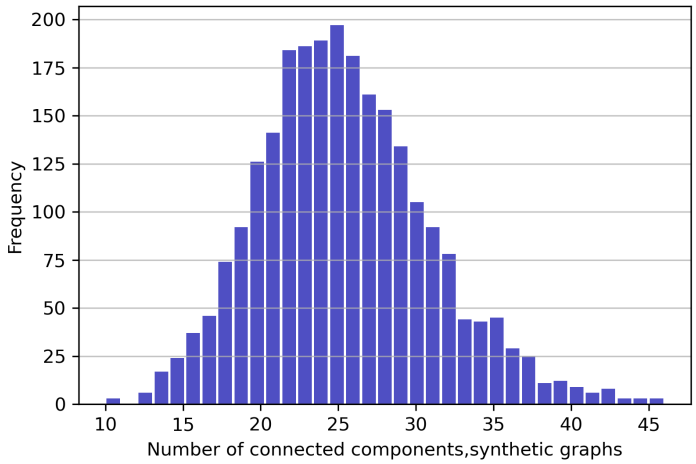


**Figure 5.5:** Number of connected components

We have also observed that almost all the generated graphs have one giant component that includes at least 80% of the nodes in the graph. Albeit for a rigorous statement we should investigate each graph separately, we can say that we expect that because the average degree value of the real grids (that is most of the time $> 2.3$) and the Hamiltonian specification that we are using. It is important to mention that we could technically consider only the giant components of these graphs and consider that as the synthetic grid, however this will obviously introduce a strong bias in our results (especially in the average bus-type degree, that was the main reason behind our proposed model) and moreover will not solve by any mean the discrepancy between the real value clustering coefficient and the generated ones. Because of this and the incapability of the model to capture the real triangle count of the grids seen in Fig. 5.4, it appears clear that we need new model's specifications to solve these issues.

## 5.2 Model II: Edges-Types with Triangles and 2-Triangles

As seen in the previous section, the ERG model with just the different type of edges counts (5.9) does not capture the higher clustering coefficient value that is a characterizing property of the power networks. In the ERG literature to model networks with such properties the number of triangles, or functions of this number, have been included in the Hamiltonian.

This seems natural recalling that the global clustering coefficient is calculated as the ratio of the total number of triangles to the total number of 2-paths in the graph. In fact, one of the

earliest ERG model for clustered graphs is the so called *Strauss's model of transitive networks* [38], which includes as observables in the Hamiltonian the number of edges and the number of triangles. However, this model is known to be prone to degeneracy and ultimately this specification of the Hamiltonian has been proved to lead to degenerate behaviours by Chatterjee and Diaconis [30]. Snijders et al. [39] have introduced a new class of models that exhibit the desired transitivity and are not prone to degeneracy: the main idea is to substitute the number of triangles with other statistics that are functions of the $k-$triangles, which we have introduced already in Chapter 4.

Let $T_1, T_2, T_3, \dots$ define the number of $1-, 2-, 3-, \dots$ triangles in a graph. Then, the proposed statistic takes the name of "*Alternating k-triangles*" and has the following form:

$$u_\lambda^{(t)} = 3T_1 - \frac{T_2}{\lambda} + \frac{T_3}{\lambda^2} - \frac{T_4}{\lambda^3} + \cdots + (-1)^{n-3}\frac{T_{n-2}}{\lambda^{n-3}} \tag{5.10}$$

The intuitive idea is that by introducing a statistic based on the $k-$triangles counts with alternating signs and decreasing weight one can model correctly the transitivity without leading to nearly fully connected (nor nearly empty) graphs, since the alternating terms will compensate each other.

### 5.2.1 Hamiltonian definition

In Section 4.2.1 we showed that most of the "normotriangular" power networks exhibit a $k-$triangles count that is approximately 0 for $k > 2$. For this reason and for computational simplicity, we decided to consider a model that, besides the six edge-types counts, includes as separate terms also the number of triangles $T_1$ and that of 2-triangles $T_2$. The resulting Hamiltonian specification is:

$$H = \beta_{PP}|E_{PP}| + \beta_{PL}|E_{PL}| + \beta_{PI}|E_{PI}| + \beta_{LL}|E_{LL}| + \beta_{LI}|E_{LI}| + \beta_{II}|E_{II}| + \beta_{1t}T_1 + \beta_{2t}T_2. \tag{5.11}$$

We will further impose that $\beta_{1t}$ and $\beta_{2t}$ have alternating signs to obtain the same compensating effect that was achieved by using alternating $k-$triangles. For sake of brevity, we will refer to this model as *ET*-model.

Notice that albeit the exact reason why this specification gives rise to transitivity without degeneracy is not yet understood, heuristics and simulations (as we will see in Section 5.3) suggest that this model can achieve the desired clustering properties. It is also worth mentioning that, to the best of our knowledge, in the context of sparse graph there is no ERG specification

that is rigorously proven to model transitivity correctly. Specifically, in [30] the authors prove using graphons theory this result for a model with number of edges, 2-paths and triangles as sufficient statistics with alternating signs however this result holds only for dense graphs.

### 5.2.2 Parameter Estimation for Model II: first proposal

Moving away from the model with only the edge-counts, we lose the tractable form of the partition function described in Theorem 1, thus we need new tools for the estimation of the parameters $\beta$. An approximation of the solution for the model with just Edges and Triangles count can be obtained using mean-field techniques [40], however this approximation has not rigorous foundations and moreover it requires that the considered graphs are dense as an assumption.

In the recent years the prominent approach for parameter estimation for ERGMs is the so called **Markov Chain Monte-Carlo Maximum Likelihood** (MCMC-MLE in the following) introduced by Geyer [41] in 1991. We will now give the rigorous statement of the convergence theorem of the MCMC-MLE in the general case [42]:

**Theorem 2** ([42]). *Let $\mathcal{K} = \{k_\beta : \beta \in \mathrm{B}\}$ be a family of non-negative functions depending on a unknown parameter $\beta$ which belongs to the space of parameters $\mathrm{B}$, integrable with respect to a measure $\mu$ such that none is integrating to $0$. Let the integrals be denoted by $c(\beta) = \int k_\beta d\mu$. Let $\mathcal{F} = \{f_\beta : \beta \in \mathrm{B}\}$ be the normalized family associated to $\mathcal{K}$, with $f_\beta$ defined by*

$$f_\beta = \frac{k_\beta}{c(\beta)}. \tag{5.12}$$

*For any such family, samples $X_1, X_2, \ldots$ from $P_\beta$ can be generated through Metropolis-Hastings algorithm without knowledge of the normalizer $c(\beta)$. Then, under continuity of the maps $\beta \to k_\beta$, the **Monte Carlo Maximum Likelihood** $l_{n,\beta}$ corresponding to an observation $x$ defined by*

$$l_{n,\beta} = \log\left\{\frac{k_\beta(x)}{k_\psi(x)}\right\} - \log\left\{\mathbf{E}_{n,\psi}\Big(\frac{k_\beta(X)}{k_\psi(X)}\Big)\right\}, \tag{5.13}$$

*where $\psi$ is an arbitrary fixed parameter point and $\mathbb{E}_{n,\psi}$ indicates the "empricial" expectation, that is*

$$\mathbf{E}_{n,\psi}(g(X)) = \frac{1}{n}\sum_{i=1}^{n} g(X_i) \tag{5.14}$$

*converges to the log-likelihood ratios* $l(\beta)$ *in the limit* $n \to \infty$

$$l(\beta) = \log\left\{\frac{k_\beta(x)}{k_\psi(x)}\right\} - \log\left\{\frac{c(\beta)}{c(\psi)}\right\}. \tag{5.15}$$

The details of the proof are quite technical and beyond the scope of this project, however we highlight some important details: first of all by using this method we don't need to compute nor estimate the normalizer $c(\beta)$ that within the ERG framework corresponds to the partition function $Z(\beta)$. Moreover this method guarantees theoretical convergence starting from any starting point, provided that the number of samples is sufficiently high. From a practical point of view however, an ill starting point $\psi$ (i.e., a point whose distance from the solution is too big) can make the algorithm not converge, thus the approach suggested by Geyer and used also, for example, in the R-package **ERGM** [43] is to reiterate the procedure many times using the results of the previous step as starting parameters.

Albeit coming with good theoretical properties, this approach can be computationally unfeasible in our context when the number of nodes becomes too big, because for each time the procedure is iterated the chain is required to take the samples after reaching the mixing time, textiti.e., the number of steps after which a chain is said to be close to the steady state distribution, and this could be not feasible for larger $n$ (for the ERGMs on dense graphs it has been proved that the mixing time is of the order $\mathcal{O}(n^2 \log n)$ [44], however currently there are no similar results for ERGM on sparse graphs).

A recent paper by Borisenko et al. [45] introduced a new simple and promising way to do parameters estimation for exponential family distributions, so including also the ERGMs: this method has connections with the **Persistent Constrastive Divergence** [46] commonly used to train the Restricted Boltzmann Machines and it is still based on a Markov Chain Monte Carlo algorithm with constantly updating parameters, resembling the technique known as **simulated annealing**. This new proposed algorithm takes the name **Equilibrium Expectation**, and now we will give both the pseudocode and outline some of its theoretical details.

**Algorithm 2** Equilibrium Expectation

**Input:** $G_0 = (V_0, E_0)$, original graph corresponding to the real grid; *N_step*
*Step_count* $= 0$
**for** $k = 1, \ldots, K$ **do**
  Sample a random edge $(i, j)$
  **if** $(i, j) \in E$ **then**
    Remove the edge: $E^k = E^{k-1}/(i, j)$
    Accept $G^k$ with probability $a = \min\left\{1, P(G^k)/P(G^{k-1})\right\}$
    **if** Move is accepted **then**
      *Step_count* $+= 1$
    **end if**
  **else**
    Add edge: $E^k = E^{k-1} \cup (i, j)$
    Accept $G^k$ with probability $a = \min\left\{1, P(G^k)/P(G^{k-1})\right\}$
    **if** Move is accepted **then**
      *Step_count* $+= 1$
    **end if**
  **end if**
  **if** *Step_count* $==$ *N_step* **then**
    *Step_count* $= 0$
    Update parameters according to the **Update Rule**
  **end if**
**end for**

Here *N_step* is a user defined input variable that determines the number of steps after which the parameter should be updated. This update is done according to the chosen **Update Rule**. The rule proposed by the authors in the paper is the following: let $\beta_0, \beta_1, \ldots, \beta_p$ the model's parameters and $x_1(G), x_2(G), \ldots, x_p(G)$ be the associated observables' values for graph $G$, then each parameter $\beta_i$ will be updated simultaneously in the following way:

$$\beta_i^{t+1} = \beta_i^t + \alpha \cdot \max(|\beta_i^t|, c) \cdot \text{sign}[x_i(G_0) - x_i(G^t)] \tag{5.16}$$

Where $G_0$ is the graph corresponding to the real grid, $\beta_i^{t+1}$ is the $i-$parameter at the $t+1$ update, $x_i(G^t)$ is the $i-$observable value obtained from $x_i(G^{t-1})$ after *N_step* moves of the

Metropolis-Hastings algorithm; $\alpha$ is the learning rate that must be given as an input and $c$ is a control parameter that assures that the algorithm doesn't get stuck even when the value of the parameter is close to zero.

The MLE can be computed as the average of the resulting sequence:

$$\hat{\beta}_i^{MLE} = \lim_{t\to\infty} \frac{1}{t - t_B} \sum_{j=t_B+1}^{t} \beta_i^j, \tag{5.17}$$

where $t_B$ is a burn-in time.

The results on the convergence of the method are summed up in the following theorem:

**Theorem 3.** *For a sufficiently small learning rate $\alpha$, if algorithm 2 converges, it converges to the **MLE** solution.*

*Proof.* Within the ERG family as we have seen in chapter 3 a key property is that $\hat{\beta}$ is the **MLE** if the following set of equations hold:

$$x_i(G_0) = \mathbb{E}_{\hat{\beta}}(x_i(G)) \quad \forall i \in \{1, 2, \ldots, r\} \tag{5.18}$$

With $\mathbb{E}_{\hat{\beta}}(x_i(G))$ being the average of the $i-$th observable over the graphs sampled from the chain with stationary distribution $\pi(G|\hat{\beta})$.

We cannot compute exactly $\mathbb{E}_{\hat{\beta}}(x_i(G))$, however Snijders [37] proposed to use $t-$ratios as a test to prove convergence using sample mean and standard deviation:

$$t_i = \frac{\hat{\mathbb{E}}_{\hat{\beta}}(x_i(G)) - x_i(G_0)}{\sigma_{\hat{\beta}}(x_i(G))} \tag{5.19}$$

If $|t_i| < 0.1$ for each $i$ in $\in \{1, 2, \ldots, r\}$, then it can be said that there is excellent convergence.

Algorithm 2 is said to converge when all the parameters oscillate around a mean value, that we call $\bar{\beta}$. If equation (5.19) is satisfied and the algorithm converges and has as the stationary distribution $\pi(G|\bar{\beta})$, then we say that the algorithm converges to the **MLE**. Thus we must now prove that with the same acceptance rule as the Metropolis-Hastings algorithm, the method has $\pi(G|\bar{\beta})$ as a stationary distribution.

The authors showed with computational experiments that when the algorithm is applied then the following are satisfied:

$$\sigma(\beta_i) \propto \alpha \quad \forall i \in \{1, 2, \ldots, r\} \tag{5.20}$$

Notice now that we are sampling from a exponential family distribution with uncertain parameters. Within this framework, Ceperley and Dewing [47] proved the result that we now rewrite for our specific case.

**Proposition 1.** *Assume that after each step, an estimate of the difference between the two parameters configuration corresponding to the chain states $s$ and $s'$ is available, which we shall denote as $\delta = \delta(s \to s')$. Let now $a = a(s \to s')$ be a modified acceptance probability of going from state $s$ to state $s'$, that we assume to depend only on $\delta$, and let $P(\delta_0; s \to s')$ be the probability of $\delta = \delta_0$. Then the average acceptance probability $A(s \to s')$ will be given by:*

$$A(s \to s') = \int_{-\infty}^{\infty} dP(\delta; s \to s') a(\delta) \tag{5.21}$$

*And consequently the detailed balance equation of the chain, given the desired stationary distribution, becomes:*

$$\pi(s|\bar{\beta}) T(s \to s') A(s \to s') = \pi(s'|\bar{\beta}) T(s' \to s) A(s' \to s) \tag{5.22}$$

*Where $T(s \to s')$ is the transition probability from state $s$ to state $s'$ and $\pi$ is the true (unknown) stationary distribution.*

We now define

$$\Delta = \frac{\pi(s'|\bar{\beta}) T(s' \to s)}{\pi(s|\bar{\beta}) T(s \to s')} \tag{5.23}$$

So that we can rewrite (5.22) as follows

$$A(s \to s') = \Delta \cdot A(s' \to s) \tag{5.24}$$

Notice that in our framework the process to estimate $\delta$ can be assumed to be symmetric in $s$ and $s'$, that means $P(\delta_0; s \to s') = P(-\delta_0; s' \to s)$; using this and equality (5.21) we can rewrite (5.24) in the following way:

$$\int_{-\infty}^{\infty} dP(\delta; s \to s') [a(\delta) - \Delta a(-\delta)] = 0 \tag{5.25}$$

Both $P(\delta_0; s \to s')$ and $\Delta$ are unknown, however we can assume that the differences $\delta$ are distributed according to a Normal distribution, a thing that is surely true if we have $t \to \infty$ because of the Central Limit Theorem. With this assumption and by assuming $\langle \delta \rangle = \Delta$ the

authors showed that a solution for equation (5.25) is given by the following $a$

$$a(\delta;\sigma) = \min\left\{1, \exp\left(-\delta - \sigma^2/2\right)\right\} \tag{5.26}$$

That with our standard notation can be rewritten as

$$a(G, G';\sigma) = \min\left\{1, \frac{P(G')}{P(G)} \exp\left(-\sigma^2/2\right)\right\} \tag{5.27}$$

But now from equation (5.20), if we chose a sufficiently small stepsize $\alpha$, $\sigma^2 \to 0$ and the term $\exp\left(-\sigma^2/2\right)$ can be neglected, obtaining the same acceptance rule as for the classic Metropolis-Hastings algorithm. Thus if the algorithm converges, it converges to the MLE. □

### 5.2.3 Theoretical results for Model II

Generating even just one connected graph using the ERG models that we have presented so far is quite challenging even for smaller network sizes, and becomes progressively harder as the number of nodes increases. A possible way to overcome this problem was proposed by Grey *et al.* [48]: in their paper they propose a modification of the Metropolis-Hastings algorithm for random graphs in which a constraint is imposed on the space of the states that can be reached by the chain based on the desired property that the Graphs should exhibit (for example, the connectivity). We will now report here how this algorithm works in practice:

**Algorithm 3** Connected graph generation [48]

Start from $G^0 = (V^0, E^0)$, connected graph
**for** $k = 1 \dots K$ **do**
  Generate a random edge $(i, j)$
  **if** $(i, j) \in E$ **then**
    Remove the edge: $E^k = E^{k-1} \setminus (i, j)$
    **if** $G^k$ is connected **then**
      Accept $G^k$ with probability $a = \min\left\{1, P(G^k)/P(G^{k-1})\right\}$
    **else**
      Reject $G^k$
    **end if**
  **else**
    Add edge: $E^k = E^{k-1} \cup (i, j)$
    Accept $G^k$ with probability $a = \min\left\{1, P(G^k)/P(G^{k-1})\right\}$
  **end if**
**end for**

The algorithm starts in a connected state and for each addition-move it behaves like the Metropolis-Hastings algorithm, whereas for removing moves it first checks if the move disconnects the graph and if so the move is rejected. The acceptance probability in both case is the same as for the M-H. This is due to the following remark:

**Remark 1.** *Within the framework of 3 we want to sample from the space of connected graphs. The acceptance probability thus is given by*

$$a = \min\left\{1, \frac{P(G^k \mid G^k \text{ is connected})}{P(G^{k-1} \mid G^{k-1} \text{ is connected})}\right\}. \tag{5.28}$$

*The ratio given here is intractable since we cannot compute* $P(G \mid G \text{ is connected})$*, however since all the evaluated moves in the algorithm must preserve connectivity, i.e.,* $P(G, G \text{ is connected}) = P(G), \forall G$ *and since* $P(G \text{ is connected})$ *is constant over the ensemble, we can rewrite:*

$$\begin{aligned} a &= \min\left\{1, \frac{P(G^k, G^k \text{ is connected})}{P(G^{k-1}, G^{k-1} \text{ is connected})} * \frac{P(G^{k-1} \text{ is connected})}{P(G' \text{ is connected})}\right\} \\ &= \min\left\{1, P(G^k)/P(G^{k-1})\right\} \end{aligned} \tag{5.29}$$

*Obtaining the same acceptance rule as for the Metropolis Hastings*

We will now prove the following theorem:

**Theorem 4.** *Algorithm 3 produces a chain that converges to the stationary distribution* $\pi = P(G \mid G \text{ is connected})$

*Proof.* The chain produced by the Metropolis-Hastings algorithm converges to the stationary distribution if it is an aperiodic and irreducible chain. We have seen in 1 that the algorithm 3 has the same acceptance rule as the M-H and is in fact a Metropolis-Hastings on the space of connected graphs. We have now to show that the chain that it generates it's also irreducible and aperiodic. The chain is irreducible since with the given acceptance rule there are no absorbing states and moreover from each connected graph configuration it is possible to reach each other connected configuration. Let $F$ be the fully connected graph (*i.e.,* the graph with all the possible edges). Since the algorithm can add any edge with positive probability, and can delete any edge with positive probability as long as it remains in the space of connected graphs we have that for any $G$ connected graph:

$$P(G \to F) > 0 \tag{5.30}$$

$$P(F \to G) > 0 \tag{5.31}$$

Therefore the irreducibility is proven.

To prove that the chain is aperiodic, given that is irreducible, it suffices that $P(G^{t+1} = G^t) > 0$ for some $G$. By construction if the move proposed disconnects the graph the move is rejected and thus $G^{t+1} = G^t$, proving aperiodicity. Since the chain is irreducible and aperiodic it converges to the stationary distribution that by construction of the algorithm is $\pi = P(G \mid G \text{ is connected})$. □

From a computational point of view, this algorithm needs to check for the connectivity of the graph each time the proposed move consists in the removal of an edge, something that we can do by using a simple breadth first search (BFS), an algorithm that normally has complexity $\mathcal{O}(n + |E|)$ but since in our framework we are dealing with sparse graph for which $|E| \approx N$ the overall complexity of the BFS algorithm is $\mathcal{O}(n)$, making this approach feasible even for bigger grids.

Within the ERG framework, since we are imposing a constraint on the chain we need also to take into account this during the parameter estimation phase: in fact using the Algorithm 3

we have that the equilibrium distribution $\pi = P(G \mid G \text{ is connected})$, whereas the vector of parameters $\beta$ generally estimated for the ERG ensemble takes into account also the disconnected graph (as we have already seen the naïve Metropolis-Hastings algorithm with correct parameters produces, after the mixing time has been reached, graphs whose observables have on average the same values as the real ones [24], however this averaging is achieved considering both connected and disconnected graph).

In the following, we will see how the algorithm in 3 can be used in conjunction with the *Equilibrium Expectation method* described in Algorithm 2 to obtain a correct parameter estimation with a constrained state space.

Consider now the Equilibrium Expectation method described in Algorithm 2 where, instead of the standard Metropolis-Hastings algorithm, we use Algorithm 3. By doing this we are effectively putting the constraint also during the parameter estimation phase, a thing that seems intuitively reasonable considering that for an ERG with exact parameters (*i.e.,* parameters that satisfy Eq. (3.8)) it is guaranteed that after the mixing time has been reached the average value of the observables of the samples drawn from the chain would be the real one, however this averaging takes into account also the disconnected graphs, resulting in a biased result if we remove them (to get an intuition of why this is true, consider that graphs with more edges are more likely connected than those with fewer).

We write now the resulting algorithm that we propose in its more general form, *i.e.,* not specifically to ensure connectivity but to ensure that the graph stays in any Graph-space $\mathcal{S}$,$\mathcal{S} \subseteq \mathcal{G}$, $G_0 \in \mathcal{S}$, for which it holds that the function $f(G)$ = "adding an edge to $G$" goes from $\mathcal{S}$, where defined, to $\mathcal{S}$ (if we call $D_f$ the domain of $f$ we say $f : D \cap \mathcal{S} \to \mathcal{S}$) and that the function $g(G)$"removing an edge from $G$" has the image whose intersection with $\mathcal{S}$ is non-empty (we say $g : D_g \cap \mathcal{S} \to I$ with $I \cap \mathcal{S} \neq \varnothing$):

**Algorithm 4** Equilibrium Expectation for a constrained chain

Start from $G_0$, original graph corresponding to the real grid
*Step_count* $= 0$
**for** $k = 1 \dots K$ **do**
  Generate a random pair $i, j$ from $1, \dots, N$
  **if** $(i, j) \in E$ **then**
    Remove the edge: $E^k = E^{k-1} \setminus (i, j)$
    **if** $G^k \in \mathcal{S}$ **then**
      Accept $G^k$ with probability $a = \min\left\{1, P(G^k)/P(G^{k-1})\right\}$
    **else**
      Reject $G^k$
    **end if**
    **if** Move is accepted **then**
      *Step_count* $+= 1$
    **end if**
  **else**
    Add edge: $E^k = E^{k-1} \cup (i, j)$
    Accept $G^k$ with probability $a = \min\left\{1, P(G^k)/P(G^{k-1})\right\}$
    **if** Move is accepted **then**
      *Step_count* $+= 1$
    **end if**
  **end if**
  **if** *Step_count* $==$ *N_step* **then**
    *Step_count* $= 0$
    Update parameters according to the **Update Rule**
  **end if**
**end for**

For this last algorithm we prove the following theorem that is also one of the main theoretical results of our work:

**Theorem 5.** *If Algorithm 4 converges, it converges to a chain whose stationary distribution is* $\pi(G \mid G \in \mathcal{S})$

*Proof.* First we make a remark: the chain produced by Algorithm 4 moves only in $\mathcal{S}$. This is

true because the chain starts in the real graph $G_0$, and we have as assumptions that $G_0 \in \mathcal{S}$ and that adding an edge will keep the chain inside $\mathcal{S}$, the same is true for the removal after the checking step that will ensure that we are still in the set.

We can then follow the steps of the proof of Theorem 3, where instead of $\pi(s \mid \bar{\beta})$ we have $\pi(s \mid s \in \mathcal{S}, \bar{\beta})$, and we end up with equation (5.26) that we rewrite here:

$$a(\delta; \sigma) = \min \left\{1, \exp\left(-\delta - \sigma^2/2\right)\right\} \tag{5.32}$$

Now $\exp(-\delta)$ is given by $\frac{P(G^k \mid G^k \in \mathcal{S})}{P(G^{k-1} \mid G^{k-1} \in \mathcal{S})}$ and we can use the result in equation (5.29) to obtain

$$a(G, G'; \sigma) = \min \left\{1, \frac{P(G')}{P(G)} \exp\left(-\sigma^2/2\right)\right\} \tag{5.33}$$

That is the same result as for the algorithm 2 and so if the condition in (5.19) holds, then the algorithm converges to the **MLE**. □

### 5.2.4 Computational results for Model II

We present now as an example the results of the graphs sampled by the model with Hamiltonian specification as in (5.11) with parameters estimated using algorithm 4, imposing thus also connectivity. We used as the real input network again the test case "300 ieee", the same we used as an input network for the simplest model in Section 5.1.2.

The procedure we implemented followed the steps, which we summarize here:

1. First we extract all the topological and electrical information we need (whole network topology, edge-type counts, triangles count, $2-$triangles count).

2. We then compute all the parameters for the edges-only model using Eq. (3.8). We will use this as a starting point $\beta_{\mathbf{1}}$. For the triangles and 2-triangles' parameters we will use an educated guess based on another method (MCMCMLE, interpolation of previous results, MPLE,...) as a starting point $\beta_{\mathbf{2}}$.

3. With the starting point $\beta = (\beta_{\mathbf{1}}, \beta_{\mathbf{2}})$ we initialize algorithm 4, until after convergence is reached for all parameters. We memorize both the trajectories of each parameter as well as the final estimated parameters $\hat{\beta}$ computed as in equation (5.17).

4. We use $\hat{\beta}$ as the parameters for algorithm 3 in order to generate the connected graph trajectory. We make the algorithm run past the burn-in time $t_B$ and memorize all the graphs generated after $t_B$.

5. As we did in Section 5.1.2 we deal with the autocorrelation of the generated chain by thinning it with the rationale based on the hamming distance of the adjacency matrices associated with the graphs explained in Section 5.1.

For the considered network this whole procedure took 2 hours to generate 2252 weakly-correlated graphs. We report here a descriptive analysis of the method, starting with the trajectories of each of the eight considered parameters.

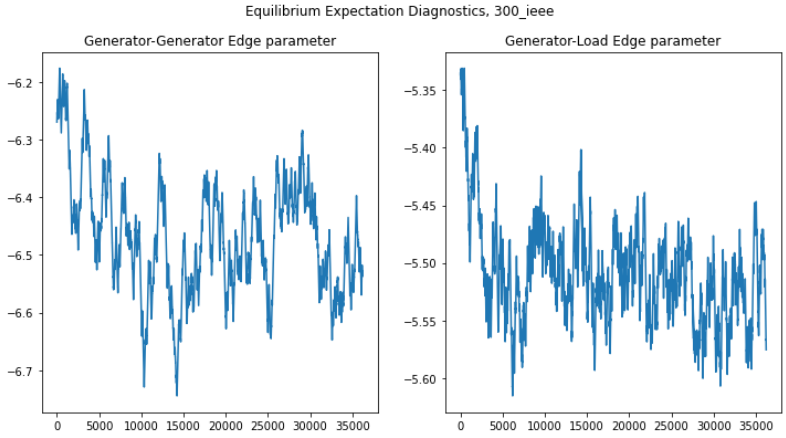


**Figure 5.6:** Generator-Generator, Generator-Load edge parameter

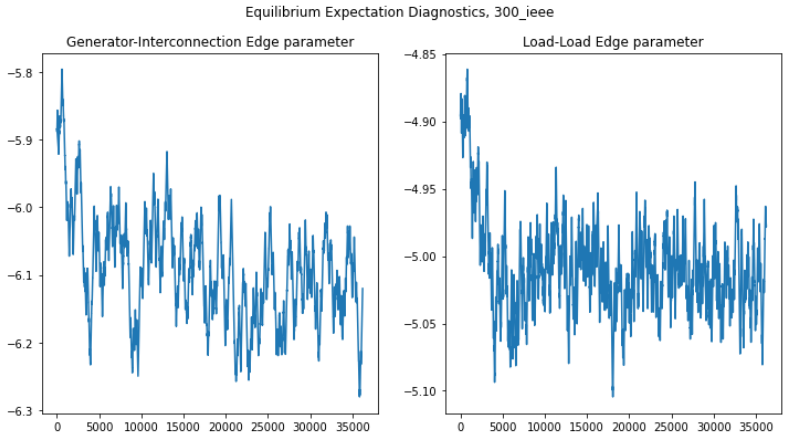


**Figure 5.7:** Generator-Interconnection, Load-Load edge parameter

In **??????** we see the behaviour of each parameters during the estimation via algorithm 4. For the edge-counts parameters we can see that after an initial phase of stable increasing or decreasing, the value starts to oscillate around the, supposedly, solution of (3.8). For both the

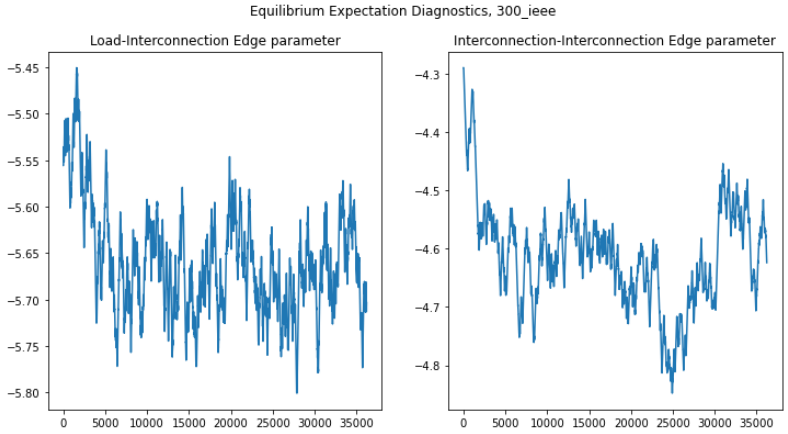


**Figure 5.8:** Load-Interconnection, Interconnection-Interconnection edge parameters

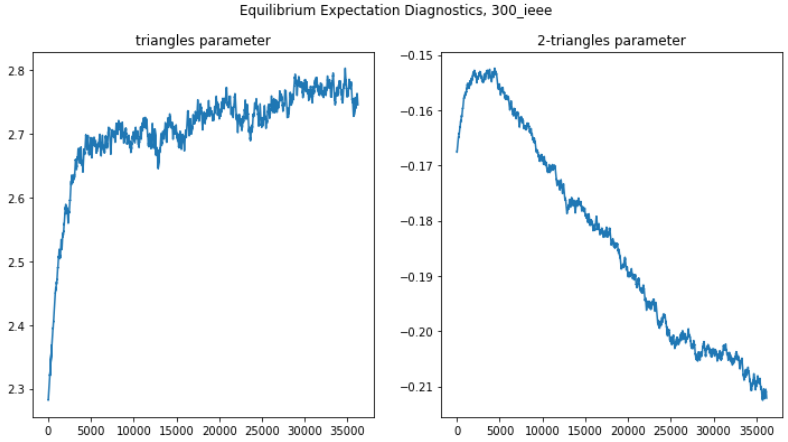


**Figure 5.9:** Triangles and 2-Triangles parameters

parameters referring to Triangles and 2-Triangles 5.9, for which we used an educated guess that thus could theoretically be far away from the real solution, we see a more steep behaviour, and in particular the 2-Triangles parameter is the last to converge. This could be explained by the inherent correlation between these two parameters, a thing that we should take into account, especially when dealing with bigger grids.

Now, like we did in Section 5.1.2 we report the plots referring to the most important properties of the grids for the scope of our research, computed on the 2252 graphs obtained with the procedure. We recall that in this case all the graphs are not only weakly-correlated with each other, but also connected.

As we can see in the figures Figs. 5.10 to 5.12 the average degrees per bus type are closely

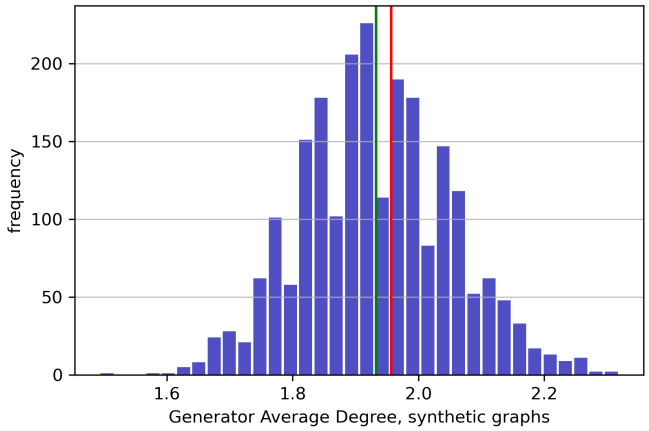


**Figure 5.10:** Generator average degree, the red line represents the real value whereas the green one represents the average among the synthetic samples

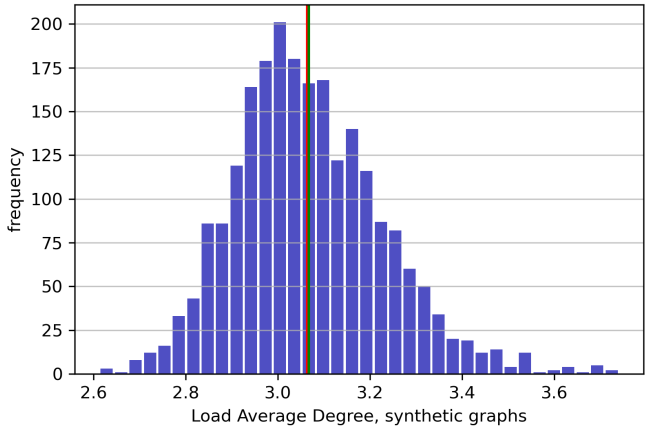


**Figure 5.11:** Load average degree, the red line represents the real value whereas the green one represents the average among the synthetic samples

captured on average by the generated graphs. In Figs. 5.13 and 5.14 we see how the graphs generated by the model are close to the real grid with respect both the triangles count and the 2−triangles count, meaning that our specification is able to model correctly the typical transitivity of the power networks without degeneracy. Moreover this further confirms that the parameters estimated within Algorithm 4 are indeed correct for this Hamiltonian specification with the constraint of connectivity. In Fig. 5.14 the distribution of the 2−triangles is skewed compared to the ones of other observables, however as already stated the average value com-

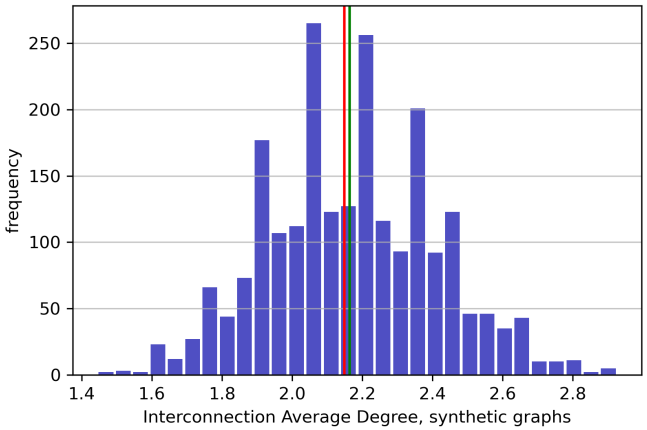


**Figure 5.12:** Interconnection average degree, the red line represents the real value whereas the green one represents the average among the synthetic samples

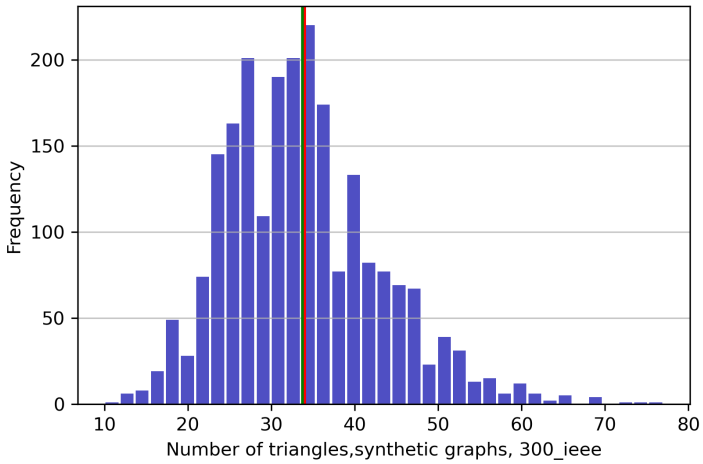


**Figure 5.13:** Number of triangles

puted on the generated graphs (highlighted in green) and the real value (highlighted in red) are almost equal, again proving that our estimation method converges to the correct set of parameters also for ones relative to the triangles.

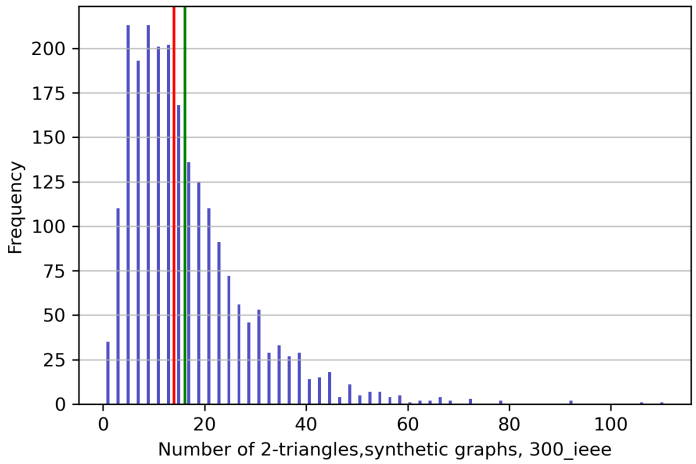


**Figure 5.14:** Number of $2-$triangles

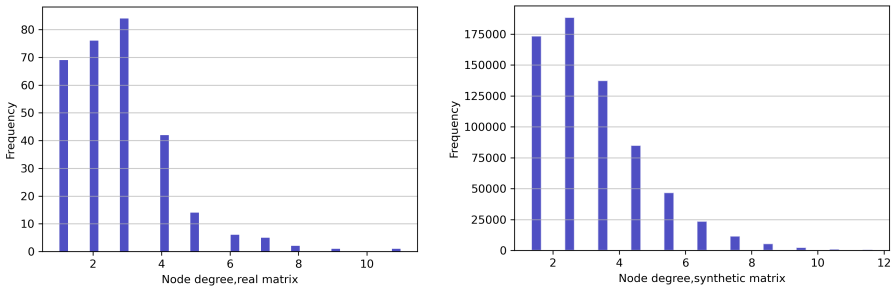


**Figure 5.15:** Comparison between real degree distribution and degree distribution among synthetic grids

In Fig. 5.15 we have a comparison between the distribution of the degrees in the real graph, and the degree distribution among all the generated graphs. The two distributions are really similar with each other, and this represent another proof of the goodness of fit of our model.

We chose also to compare our synthetic networks to the real one with respect to two of the main properties of the real grids, that we have not included in our Hamiltonian specification nor in our estimation algorithm: the average path length and the algebraic connectivity.

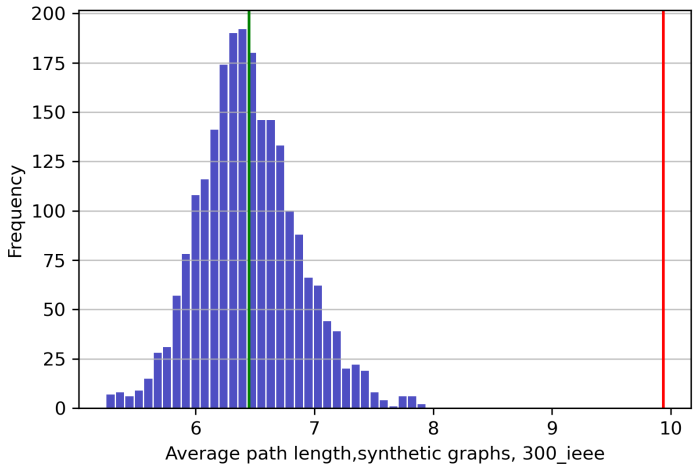


**Figure 5.16:** Average path length, the red line represents the real value whereas the green one represents the average among the synthetic samples

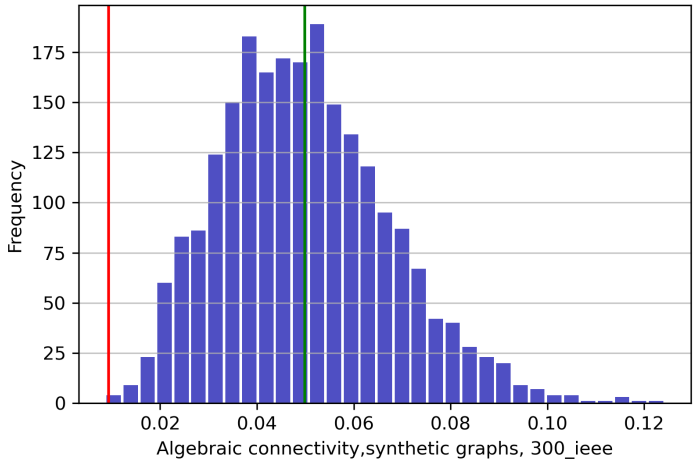


**Figure 5.17:** Algebraic connectivity, the red line represents the real value whereas the green one represents the average among the synthetic samples

The histogram in Fig. 5.16 shows that the mean value of the average shortest path length for the generated graphs (in green) is smaller than the real value (in red), even if the order of magnitude is the same. Similarly, as we can see in Fig. 5.17, the algebraic connectivity of the synthetic graphs has an average value (in green) that is higher than the one of the grid of reference (in red), which is located on the tail of the distribution. Even if these properties are not

accurately captured by our model, we should highlight that the average values obtained on the generated samples can be regarded as realistic for both the APL and the algebraic connectivity by the standards used for example in [2] to assess the realism of a synthetic grid.

## 5.3 Comparison with the real grids

We propose here a comparison done with respect to the principal properties of the network with the real ones.

| Networks | $N$ | $\langle k_P \rangle$ | $\langle k_L \rangle$ | $\langle k_I \rangle$ | APL | $\lambda_2$ | $\lvert T \rvert$ | $\lvert T_2 \rvert$ |
|---|---|---|---|---|---|---|---|---|
| 118 ieee | 118 | 3.56 | 2.5 | 3.25 | 6.3 | 0.027 | 23 | 11 |
| Average synth 118 | 118 | 3.61 | 2.52 | 3.33 | 4.67 | 0.1 | 24 | 13 |
| 300 ieee | 300 | 1.95 | 3 | 2.15 | 9.93 | 0.0093 | 34 | 14 |
| Average synth 300 | 300 | 1.93 | 3.06 | 2.16 | 6.44 | 0.049 | 33.7 | 16 |
| 1354 PEGASE | 1354 | 2.58 | 1.08 | 2.53 | 11.15 | 0.005 | 87 | 14 |
| Average synth 1354 | 1354 | 2.53 | 1.16 | 2.51 | 9.56 | 0.022 | 84 | 19 |

**Table 5.1:** Comparison of our model's results and real grids

The observations done in Section 5.2.4 apply also to the other considered grids. Our generated graphs are close to the real ones from both the node and edge count, the average degrees per bus type and the triangles and $2-$triangles count. It is less reliable, albeit still within a realistic range, for the average shortest path length and the algebraic connectivity. We remark also here that all our graphs come with a correct bus type assignment because of the Hamiltonian specification that we used (5.11). With our current implementation, er were not able to use our method to generate bigger grids (more than 2500 nodes) because of the excessive mixing time of the generated chain. We are not currently able to determine the mixing time of our method since, as stated also before, there are currently no studies on the mixing time of sparse ERGs. Moreover, the constraint that we are imposing on the chain will likely lead to an even worse mixing of the chain. In the future works, we should find a solution to deal with bigger grids (as the one that we will propose in the conclusion), and also give rigorous results on the mixing of the chains generated by our algorithm.

# 6 Conclusion

In this thesis we have analyzed the topological properties of the power grids seen from a complex network perspective, in order to propose a new model of the ERG family to generate synthetic grids able to mimic closely the principal characteristics of a real power system.

We have provided an in-depth review of the current literature about modeling of power system as graphs, highlighting the fundamental aspects of each approach. The lack of models well founded from a rigorous mathematical and statistical point of view convinced us to propose new specification within the Exponential Random Graphs family, the main properties of which we have explained in detail. The ERG framework allows to use Markov-Chain Monte-Carlo models for the generation of the grids

Within this family of models we have proposed a new simple specification in 5.9, that belongs a generalized class of ERG models, for which we have also proved the existence of a closed from expression for the partition function. This model generates synthetic networks with a realistic bus type assignment, however is not able to generate graphs coherent to the real grids with respect to some of the essential properties of the real power system, namely connectivity and transitivity.

For this reason we improved our model specification and found new estimation methods: for the transitivity we added in the Hamiltonian as observables the triangle count and the 2−triangle count. By doing this, we were not able anymore to have a closed form expression for the partition function of the model, thus a method to estimate the parameters was needed. We explored the methods available in the literature, especially the ones that used Markov-Chain

Monte Carlo algorithms, and we ended up founding a promising approach in recent the work of Borisenko *et al.* [45]. We combined the parameters estimation algorithm they proposed with the work done by Grey *et al.* [48] on MCMC methods for a constrained chain in order to be able to generate samples from a chain constrained into the space of connected graphs. The main theoretical result of this thesis 5 is indeed the proof that the algorithm resulting from the combination of the works done in [45] and [48] results in a method to estimate correctly the parameters of an Exponential Random Graph with any Hamiltonian specification and with a major constraint imposed on the graphs that form the ensemble (in our case, the constraint of connectivity).

This result not only allowed us to obtain samples that closely mimic the properties included in the Hamiltonian as observables, but it opens also a lot of possibilities in other fields of research which use complex networks as the main modeling tool, for example in the Social Network analysis. In fact this model can be used for a wide class of ERG models, bypassing the limitations imposed by the ERG family by constraining the chain into the space of the desired graphs, allowing for a great flexibility (as we said before, imposing connectivity is just one of the possible usage of our method).

The synthetic grids that we generated were realistic also with respect to topological characteristics that were not included in the Hamiltonian, even if in a less accurate way.

To obtain a more complete result from a mathematical point of view, we should investigate the mixing time of the chain in our model. The literature on mixing time of dense ERGMs is quite rich [44, 30, 49], on the converse there are no specific studies on the mixing time of sparse ERGMs, like the one that we are considering in this thesis. This is due to the fact that the works done on the dense graphs largely rely on the theory of graph limits or graphons. There are currently very few research about asymptotic behaviour of sparse graphs, however it is a research field that is growing in the latter years, and in fact very recent articles like the one of Cook and Dembo [50] seem promising to lay the foundations of a rigorous mixing time analysis for the sparse Exponential Random Graph models.

Since our method is able to estimate multiple uncorrelated samples, but, with the current implementation, does not scale well for bigger grids ($N > 3000$) we propose as a possible future improvement the development of a procedure that combines our algorithm 4 with a rationale similar to the one behind the SDET method to generate synthetic grids [21]: we can use our model to generate multiple realistic synthetic grids of small-medium size and then reassemble them into bigger topologies by adapting the method proposed in [21]. In order to do so in a proper way, we should also generate the electrical and geographical properties for our synthetic

grids. This is particularly tricky since real geographical data about power systems is restricted and to the best of our knowledge there are not publicly available collections of grids' testcases with geographical information.

Even if the data was available, it is our believe that new algorithms should be needed to obtain realistic grids from a topological, electrical and geographical standpoint starting from the correct topologies that we provide with our method. This is probably the most important improvement that we should consider for our work.

# Bibliography


[1] R. Espejo, S. Lumbreras, and A. Ramos, "A complex-network approach to the generation of synthetic power transmission networks," *IEEE Systems Journal*, vol. 13, no. 3, pp. 3050–3058, 2019.

[2] H. Sadeghian and Z. Wang, "AutoSynGrid: A MATLAB-based toolkit for automatic generation of synthetic power grids," *International Journal of Electrical Power & Energy Systems*, vol. 118, p. 105757, 2020.

[3] A. Birchfield and T. Overbye, "Planning sensitivities for building contingency robustness and graph properties into large synthetic grids," in *Proceedings of the Annual Hawaii International Conference on System Sciences*. Hawaii International Conference on System Sciences, 2020. [Online]. Available: https://doi.org/10.24251/hicss.2020.386

[4] A. J. Wood, B. F. Wollenberg, and G. B. Sheblé, *Power generation, operation, and control*. John Wiley & Sons, 2013.

[5] Z. Wang, R. Thomas, and A. Scaglione, "Generating Random Topology Power Grids," *Proceedings of the 41st Annual Hawaii International Conference on System Sciences (HICSS 2008)*, pp. 183–183, 2008.

[6] R. Albert and A.-L. Barabási, "Statistical mechanics of complex networks," *Reviews of Modern Physics*, vol. 74, no. 1, jan 2002. [Online]. Available: https://doi.org/10.1103%2Frevmodphys.74.47

[7] Z. Wang, M. H. Athari, and S. H. Elyas, "Statistically Analyzing Power System Network*," *2018 IEEE Power & Energy Society General Meeting (PESGM)*, vol. 00, pp. 1–5, 2018.

[8] Z. Wang and S. H. Elyas, "On the scaling property of power grids," 01 2017.

[9] B. Carreras, V. Lynch, I. Dobson, and D. Newman, "Critical points and transitions in an electric power transmission model for cascading failure blackouts," *Chaos (Woodbury, N.Y.)*, vol. 12, pp. 985–994, 01 2003.

[10] M. Parashar, J. Thorp, and C. Seyler, "Continuum modeling of electromechanical dynamics in large-scale power systems," *Circuits and Systems I: Regular Papers, IEEE Transactions on*, vol. 51, pp. 1848 – 1858, 10 2004.

[11] D. J. Watts and S. H. Strogatz, "Collective dynamics of 'small-world' networks," *Nature*, vol. 393, no. 6684, pp. 440–442, 1998.

[12] Z. Wang, A. Scaglione, and R. J. Thomas, "Generating statistically correct random topologies for testing smart grid communication and control networks," *IEEE Transactions on Smart Grid*, vol. 1, no. 1, pp. 28–39, 2010.

[13] Z. Wang, S. H. Elyas, and R. J. Thomas, "A novel measure to characterize bus type assignments of realistic power grids," in *2015 IEEE Eindhoven PowerTech*, 2015, pp. 1–6.

[14] S. H. Elyas and Z. Wang, "Improved Synthetic Power Grid Modeling With Correlated Bus Type Assignments," *IEEE Transactions on Power Systems*, vol. 32, no. 5, pp. 3391–3402, 2015.

[15] S. Soltan and G. Zussman, "Generation of Synthetic Spatially Embedded Power Grid Networks," *2016 IEEE Power and Energy Society General Meeting (PESGM)*, pp. 1–5, 2016.

[16] S. Soltan, A. Loh, and G. Zussman, "A Learning-Based Method for Generating Synthetic Power Grids," *IEEE Systems Journal*, vol. 13, no. 1, pp. 625–634, 2017.

[17] 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*, vol. 10, no. 8, p. 1233, 2017.

[18] A.-L. Barabási and R. Albert, "Emergence of scaling in random networks," *Science*, vol. 286, no. 5439, pp. 509–512, oct 1999. [Online]. Available: https://doi.org/10.1126%2Fscience.286.5439.509

[19] 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 Transactions on Power Systems*, vol. 32, no. 4, pp. 3258–3265, 2017.

[20] 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., Dec. 2018. [Online]. Available: https://doi.org/10.2172/1524091

[21] S. J. Young, Y. Makarov, R. Diao, R. Fan, R. Huang, J. O'Brien, M. Halappanavar, M. Vallem, and Z. H. Huang, "Synthetic power grids from real world models," in *2018 IEEE Power & Energy Society General Meeting (PESGM)*, 2018, pp. 1–5.

[22] 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," 2015.

[23] S. J. Young, Y. Makarov, R. Diao, M. Halappanavar, M. Vallem, R. Fan, R. Huang, J. O'Brien, and Z. H. Huang, "Topological power grid statistics from a network-of-networks perspective," in *2018 IEEE Power & Energy Society General Meeting (PESGM)*, 2018, pp. 1–5.

[24] A. Fronczak, "Exponential random graph models," 2012. [Online]. Available: https://arxiv.org/abs/1210.7828

[25] R. Solomonoff and A. Rapoport, "Connectivity of random nets," *The bulletin of mathematical biophysics*. [Online]. Available: https://doi.org/10.1007/BF02478357

[26] P. Erdös and A. Rényi, "On random graphs. i: Publicationes," *Mathematics*, vol. 6, pp. 290–297, 1959.

[27] P. W. Holland and S. Leinhardt, "An exponential family of probability distributions for directed graphs," *Journal of the American Statistical Association*, vol. 76, no. 373, pp. 33–50, 1981. [Online]. Available: http://www.jstor.org/stable/2287037

[28] O. Frank and D. Strauss, "Markov graphs," *Journal of the American Statistical Association*, vol. 81, no. 395, pp. 832–842, 1986. [Online]. Available: http://www.jstor.org/stable/2289017

[29] T. M. Cover and J. A. Thomas, *Elements of Information Theory 2nd Edition (Wiley Series in Telecommunications and Signal Processing)*, 2006.

[30] S. Chatterjee and P. Diaconis, "Estimating and understanding exponential random graph models," 2011. [Online]. Available: https://arxiv.org/abs/1102.2650

[31] S. Babaeinejadsarookolaee, A. Birchfield, R. D. Christie, C. Coffrin, C. DeMarco, R. Diao, M. Ferris, S. Fliscounakis, S. Greene, R. Huang, C. Josz, R. Korab, B. Lesieutre, J. Maeght, T. W. K. Mak, D. K. Molzahn, T. J. Overbye, P. Panciatici, B. Park, J. Snodgrass, A. Tbaileh, P. Van Hentenryck, and R. Zimmerman, "The power grid library for benchmarking ac optimal power flow algorithms," 2019. [Online]. Available: https://arxiv.org/abs/1908.02788

[32] R. D. Zimmerman and C. E. Murillo-Sánchez, "Matpower user's manual," Oct. 2020. [Online]. Available: https://doi.org/10.5281/zenodo.4074122

[33] G. Van Rossum and F. L. Drake Jr, *Python reference manual*. Centrum voor Wiskunde en Informatica Amsterdam, 1995.

[34] T. pandas development team, "pandas-dev/pandas: Pandas," Feb. 2020. [Online]. Available: https://doi.org/10.5281/zenodo.3509134

[35] R. Christie, "Power system test archive." [Online]. Available: https://www.ee.washington.edu/research/pstca

[36] C. Josz, S. Fliscounakis, J. Maeght, and P. Panciatici, "Ac power flow data in matpower and qcqp format: itesla, rte snapshots, and pegase," 2016. [Online]. Available: https://arxiv.org/abs/1603.01533

[37] T. Snijders, "Markov chain monte carlo estimation of exponential random graph models," *Journal of Social Structure*, vol. 3, 06 2002.

[38] D. Strauss, "On a general class of models for interaction," *SIAM Review*, vol. 28, no. 4, pp. 513–527, 1986. [Online]. Available: http://www.jstor.org/stable/2031102

[39] T. A. B. Snijders, P. E. Pattison, G. L. Robins, and M. S. Handcock, "New specifications for exponential random graph models," *Sociological Methodology*, vol. 36, no. 1, pp. 99–153, 2006. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1111/j.1467-9531.2006.00176.x

[40] J. Park and M. E. J. Newman, "Solution for the properties of a clustered network," *Physical Review E*, vol. 72, no. 2, aug 2005. [Online]. Available: https://doi.org/10.1103%2Fphysreve.72.026136

[41] C. J. Geyer, "Markov chain monte carlo maximum likelihood," 1991. [Online]. Available: https://hdl.handle.net/11299/58440

[42] ——, "On the convergence of monte carlo maximum likelihood calculations," *Journal of the Royal Statistical Society. Series B (Methodological)*, vol. 56, no. 1, pp. 261–274, 1994. [Online]. Available: http://www.jstor.org/stable/2346044

[43] 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," *Journal of Statistical Software*, vol. 24, no. 3, pp. 1–29, 2008.

[44] S. Bhamidi, G. Bresler, and A. Sly, "Mixing time of exponential random graphs," 2008. [Online]. Available: https://arxiv.org/abs/0812.2265

[45] A. Borisenko, M. Byshkin, and A. Lomi, "A simple algorithm for scalable monte carlo inference," 2019. [Online]. Available: https://arxiv.org/abs/1901.00533

[46] L. Younes, "Estimation and annealing for Gibbsian fields," *Annales de l'I.H.P. Probabilités et statistiques*, vol. 24, no. 2, pp. 269–294, 1988. [Online]. Available: http://www.numdam.org/item/AIHPB_1988__24_2_269_0/

[47] D. M. Ceperley and M. Dewing, "The penalty method for random walks with uncertain energies," *The Journal of Chemical Physics*, vol. 110, no. 20, pp. 9812–9820, may 1999. [Online]. Available: https://doi.org/10.1063%2F1.478034

[48] C. Gray, L. Mitchell, and M. Roughan, "Generating connected random graphs," *Journal of Complex Networks*, vol. 7, no. 6, pp. 896–912, 03 2019. [Online]. Available: https://doi.org/10.1093/comnet/cnz011

[49] G. Bresler, D. Nagaraj, and E. Nichani, "Metastable mixing of markov chains: Efficiently sampling low temperature exponential random graphs," 2022. [Online]. Available: https://arxiv.org/abs/2208.13153

[50] N. A. Cook and A. Dembo, "Typical structure of sparse exponential random graph models," 2022. [Online]. Available: https://arxiv.org/abs/2208.06397

# Acknowledgments

Ringraziare tutte le persone che meriterebbero di essere ringraziate in modo adeguato potrebbe essere quasi impossibile. Quindi inizierò dalle parti più facili e più importanti. Ringrazio la mia famiglia: grazie a mia mamma senza la quale sarei crollato centinaia di volte, che mi ha sostenuto e coccolato quando ne avevo più bisogno e mi ha insegnato a provare ad essere la versione migliore di me stesso ogni giorno. Grazie a mio papà a cui fin da piccolo dico con orgoglio di assomigliare e che proprio per questo mi conosce meglio di chiunque altro, mi hai insegnato come ci si rialza e soprattutto a non prendersi troppo sul serio, anche quando le cose sembrano andare male. Grazie a mia sorella, con cui credo di avere il legame più sincero, prezioso e bello che io potessi desiderare, e che da sempre fa il tifo per me (e io per lei!). Grazie alla nonna Mariucci, che mi vizia da 24 anni e mi ha insegnato ad avere misura nella vita, ad essere grato per le fortune che abbiamo e ad essere orgogliosi della nostra bellissima famiglia. Grazie alla nonna Angela, che oltre a ricordarsi le date dei miei esami meglio di me mi ha insegnato a lottare e a tirare fuori la grinta nei momenti importanti. Grazie alla zia Anna, allo zio Vittorio e alla zia Elena, i migliori zii che io potessi desiderare, da cui sono stato coccolato in ogni modo possibile. Grazie allo zio Dario (senza il quale probabilmente non sarei qui), alla zia Paola e alla zia Carla e alle bellissime cene e pranzi dei cugini.

Passando agli amici premetto che sarà difficile ringraziarvi tutti a dovere ma anche qui ci proverò, nel caso non offendetevi. Partiamo da casa: grazie ai miei amici a Codroipo, Willy, Sam, Chiarotz, Panda, Francisco, Gabro, Michele detto il Banelli e perfino Michele detto il Valeo (che mi ha anche insegnato come si fanno i ringraziamenti in una tesi!). Mi avete regalato una seconda famiglia (che forse era meglio tenere segreta) da cui tornare. Grazie ad Elisa, che non si sa come mai rimane ancora mia amica nonostante mi sopporti e supporti da una vita, e grazie alla quale ho imparato a conoscere tutta la mappa del parco delle Risorgive.

Grazie a Gab e Mart, le due persone che posso dire essere più vicine a dei fratelli per me. Il fatto che siamo ancora assieme dopo così tanto tempo e dopo tutte quello che abbiamo passato mi fa sperare che ci saremo sempre luno per l'altro. Prima o poi sono sicuro che il nostro canale youtube farà il botto! Grazie ad Aurora, sei riuscita a mantenere in equilibrio il nostro gruppo e sei sempre presente per ciascuno di noi, so che un giorno potrò andare in giro vantandomi semplicemente perché ti conosco. Grazie ad Ele che ci ha sempre ispirato a puntare in alto e

che, purtroppo, mi ha anche insegnato come si parla in corsivo. Grazie a Susanna, che sempre e comunque mi ha spronato e spinto fuori dalla mia zona di comfort.

Non posso poi non ringraziare chi mi ha accolto (e a volte raccolto) nella bella città di Padova: ringrazio quindi Fez, Paolo e Ilenia per avermi educato alla vita padovana a suon di aperitivi molesti, rumori molesti e anche odori molesti. Ringrazio Hatim per aver dimostrato una pazienza infinita con me e per avermi voluto sempre bene nonostante tutto. Ringrazio Pietro per avermi insegnato che anche un vegetariano può cucinare degli ottimi arrosticini, e per aver sempre pulito dietro i disastri che lasciavo in cucina. Grazie poi ad Ance con cui ho condiviso tutto (ma proprio tutto) per 4 anni e che con la sua presenza ha reso la nostra vita un pochino più dolce. Crazie a Tuba per avermi letteralmente reso una persona migliore (crazie, bravo davvero). Grazie a Giacomo per essersi trattenuto dal chiamare la polizia e per le bellissime sbustate. Grazie ad Alessio per non averci ammazzato e per averci fatto amare gli Abba. Grazie a Ghironda per aver provato ad insegnarmi come si fa un muscle-up (un giorno ce la farò maestro!). Grazie a Parme e Frau per avermi fatto amare l'Arcella e avermi introdotto alla teoria dei memi complessi.

Grazie poi al professor Formentin che ha saputo trovare sempre del tempo per me ed è riuscito nella difficile missione di convincermi a fare un periodo di studio all'estero. Grazie quindi ad Alessandro, probabilmente il miglior supervisore che potessi incontrare; è merito tuo se ho deciso di iniziare un percorso di dottorato, quindi nel caso andasse male so a chi dare la colpa (ma ti sarò sempre e comunque grato per avermi introdotto al bouldering!). Grazie a Gianmarco che non solo mi invitato a Firenze, ma mi ha anche sostenuto e aiutato nelle mie scelte.

Now I must (PEFFORZA) say thanks to all the wonderful people I met in Amsterdam. So I say grazie to Matteo, Santi, Daniel, Neda and all the others of the gang, without you, my stay in the Netherlands would have been incredibly boring and 100% less beautiful (also I would have been probably homeless for 2 weeks!).

Grazie, infine, ai miei nonni Mario e Tullio. Mi mancate ogni giorno e non avervi fisicamente qui è un grande dolore per me, ma so per certo che in qualche modo ci siete sempre vicini. Siete state delle persone meravigliose e due nonni incredibili nonostante foste così diversi. Questa tesi la dedico a voi, vi voglio tanto bene.