Chapter 2
Maximum-Entropy Ensembles of Graphs

Whereof one cannot speak, thereof one must be silent.

—Ludwig Josef Johann Wittgenstein, *Logisch-Philosophische Abhandlung*

Abstract In this chapter we describe the core method that will be used throughout the rest of the book, i.e. the construction of a *constrained maximum-entropy ensemble* of networks. This procedure requires the definition of the *entropy* of a network ensemble, the specification of structural properties to be enforced as *constraints*, the calculation of the resulting maximum-entropy *probability* of network configurations, and the maximization of the *likelihood*, given the empirical values of the enforced constraints. We describe this procedure explicitly, after giving some general motivations. In particular, we discuss the crucial importance of enforcing *local* constraints that preserve the (empirical) heterogeneity of node properties. The maximum-entropy method not only generates the exact probabilities of occurrence of any graph in the ensemble, but also the expectation values and the higher moments of any quantity of interest. Moreover, unlike most alternative approaches, it is applicable to networks that are either binary or weighted, either undirected or directed, either sparse or dense, either tree-like or clustered, either small or large. We also discuss various likelihood-based statistical criteria to rank competing models resulting from different choices of the constraints. These criteria are useful to assess the informativeness of different network properties.

2.1 Constructing Constrained Graph Ensembles: Why and How?

In Chap. 1 we already anticipated that various problems of great importance in network science may be (re)formulated in such a way that similar underlying concepts are invoked and a common toolkit is employed. In particular, we gave a series of motivations for addressing three specific problems that will be discussed in detail in

T. Squartini and D. Garlaschelli, *Maximum-Entropy Networks*,
SpringerBriefs in Complexity, https://doi.org/10.1007/978-3-319-69438-2_2

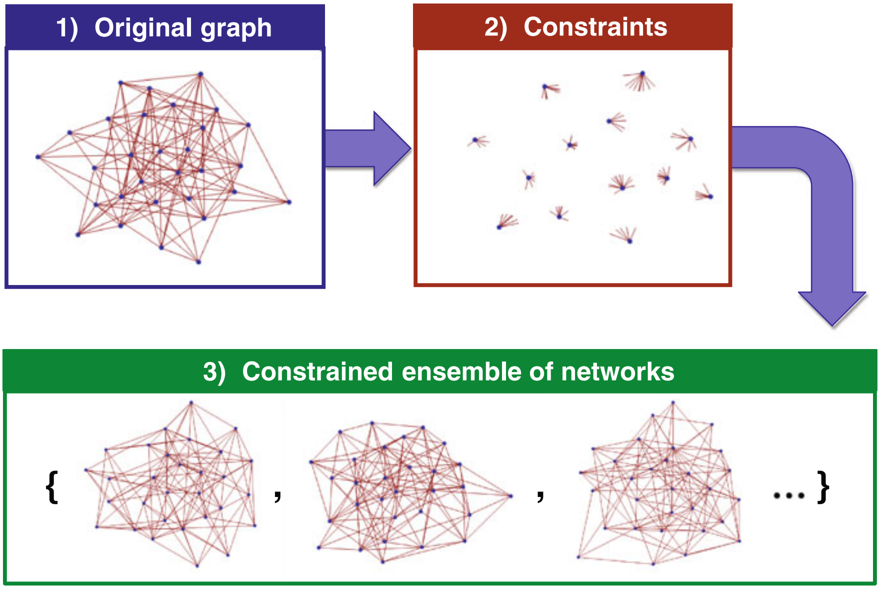

Fig. 2.1 Abstract construction of a constrained ensemble of networks. (1) First, a particular network (for instance, an observed real-world one) is considered. (2) Then, a set of topological properties (in the example shown, the different numbers of connections of nodes) is chosen as a constraint and measured on the network. (3) Finally, an ensemble of networks induced by the measured constraints is constructed according to some rule resulting in a probability distribution over the space of allowed configurations. In the problem of pattern detection (see Chap. 3), the average properties of the constrained ensemble are then compared to those of the original network in order to detect statistically significant patterns in the latter. In the problem of network reconstruction (see Chap. 4), one actually does not have empirical access to the original network, but only to a set of its properties; the procedure therefore starts at step 2) by treating these properties as constraints and then produces an ensemble of inferred possible configurations for the unknown network. Finally, in various problems in graph combinatorics (see Chap. 5), one is interested in correctly sampling and/or enumerating the configurations from the induced ensemble

the following chapters of this book, namely the detection of statistically significant structural patterns in real networks (Chap. 3), the reconstruction of networks from partial empirical information (Chap. 4) and the sampling or enumeration of graphs with specified topological properties (Chap. 5). These three different problems, while unrelated at first sight, require in fact a common framework: the construction of *an ensemble of random graphs with given constraints* [1–23]. In the case of pattern detection, the constraints represent null hypotheses used as a reference to identify empirical patterns. In the case of network reconstruction, they represent pieces of incomplete data used to infer missing information. In the case of graph combinatorics, they represent topological properties of the network configurations to be sampled or enumerated.

A pictorial representation of the construction of a constrained ensemble of graphs is given in Fig. 2.1. In general, the procedure may go through three steps: we may start from a specific (real-world) network, then measure the topological properties we want to preserve, and finally impose these properties as a constraint in the construction of the ensemble. In all the cases considered in this book, we impose that the graphs in the ensemble all have exactly the same number of nodes as the original network. It should at this point be noted that, at least conceptually, we may skip the first step and start directly with the specification of the constraints themselves (in such a case, the number of nodes in the original network should also be known, if not already evident from the constraints themselves). Whether one can actually skip the first step depends on the particular technical implementation, not on the theoretical definition of the ensemble. For instance, in certain computational pattern-detection approaches that aim at iteratively randomizing a real-world network while preserving some of its properties (explicit examples are given below in Sect. 2.1.2), one has to start from the first step. By contrast, in other cases (e.g. when only partial information is available about the original network, as in the problems considered in Chap. 4), one is forced to start from the second step. This implies that, in order to be useful for multiple purposes, 'good' ensemble constructions should be able to take (only) the values of the chosen constraints as input. Of course, this requires that such values are *graphic*,[1] i.e. realizable in at least one graph. If the constraints come from the observation of some network (including the case when they are the only information available about some unknown underlying network), their graphicality is of course always guaranteed.

In general, the third step in the construction of an ensemble of constrained graphs, i.e. the specification of a (satisfactory) graph probability, is the most challenging one. The reason is twofold, as briefly explained below.

- Firstly, not all choices of the constraints lead to equally easy ways of constructing the resulting ensemble. In fact, the most important and useful constraints turn out to be *node-specific*, which implies that the local properties of nodes have to be preserved separately. This requirement complicates the construction of the probability distribution. This point is discussed in detail in Sect. 2.1.1.
- Secondly, not all probability distributions satisfying the chosen constraints are equally acceptable from a theoretical point of view. For instance, a key requisite is that they assign the same probability to all graphs that have the same value of the constraints, because there is no reason to prefer any one such graph over any other such graph. This point is illustrated in Sect. 2.1.2 for the case of computational methods and in Sect. 2.1.3 for the case of analytical methods.

In the rest of this chapter, we explain in detail the two points above, first by highlighting the importance of imposing local constraints (Sect. 2.1.1) and then by emphasizing how most computational (Sect. 2.1.2) and analytical (Sect. 2.1.3) methods proposed in the literature fail to correctly sample the resulting ensembles. Then, in Sect. 2.2 we introduce a rigorous methodology to produce a graph probability

[1] A topological property f, where $f(\mathbf{G})$ is the value of the property in graph $\mathbf{G}$, is said to evaluate to a *graphic* (or *graphical*) value $\tilde{f}$ if there exist at least one graph $\tilde{\mathbf{G}}$ that realizes such value, i.e. for which $f(\tilde{\mathbf{G}}) = \tilde{f}$.

meeting all the desired requirements. The methodology is based on the *maximization of the entropy* subject to a set of chosen constraints (this step fixes the *functional form* of the probability distribution) and the subsequent *maximization of the likelihood* (this step fixes the *numerical values* of the probability distribution). We will see that the maximum-entropy formulation solves all the highlighted problems in an elegant and mathematically explicit way, a result that will come as a relief. This procedure represents the core of the formalism that will be used repeatedly in this book.

2.1.1 Definition and Importance of Local Constraints

To characterize the structure of a given network, arbitrarily many topological properties can be defined. Among these, the simplest and most important properties are *local* quantities, i.e. functions of only the immediate neighbourhood of each node. Let us introduce some notation to define these local properties, before discussing their importance.

A *binary undirected graph*[2] with N vertices is completely specified by a symmetric $N \times N$ *adjacency matrix* $\mathbf{A}$. The entries of the latter are such that $a_{ij} = 1$ if the vertices i and j are connected and $a_{ij} = 0$ otherwise. For each node i, the *degree* $k_i(\mathbf{A}) = \sum_{j \neq i} a_{ij}$ is defined as the number of connections of that node, and is therefore a local node-specific property. The *degree sequence* $\mathbf{k}(\mathbf{A}) = \{k_i(\mathbf{A})\}_{i=1}^{N}$ is the N-dimensional vector of degrees of all nodes.

In case of *weighted*[3] undirected graphs, a network is specified by a symmetric $N \times N$ *weight matrix* $\mathbf{W}$ where the entry w_{ij} quantifies the intensity of the link connecting nodes i and j. This includes the case $w_{ij} = 0$ corresponding to nodes i and j being not connected. Besides the degree (which is still defined as the number of connections of a node, irrespective of their intensity), another local property that can be introduced in this case is the *strength* $s_i(\mathbf{W}) = \sum_{j \neq i} w_{ij}$, defined as the sum of the weight of all links of vertex i. The *strength sequence* $\mathbf{s}(\mathbf{W}) = \{s_i(\mathbf{W})\}_{i=1}^{N}$ is the N-dimensional vector of strengths of all nodes.

[2] An *undirected* graph (or network) is a graph where no direction is specified for the edges. An undirected graph is *binary* or *simple* if each pair of nodes i and j (with $i \neq j$) is connected by at most one edge, i.e. if there are *no multiple edges* between the same two nodes. We will also assume the absence of *self-loops* (edges starting and ending at the same node) throughout the book.

[3] A *weighted graph* (or network) is a graph where links may carry different intensities. When dealing with weighted networks, throughout the book we will assume non-negative integer link weights (i.e. $w_{ij} = 0, 1, 2 \cdots + \infty$) for simplicity. This corresponds to the assumption that an indivisible unit of measure of link weights has been preliminary specified. Under this assumption, a weighted network can also be regarded as a graph that is in general not simple, i.e. where multiple links of unit weight are allowed between the same two nodes. We will still exclude the possibility of self-loops. Ideally, one may think of link weights becoming continuous as the unit of measure is chosen to be vanishingly small.

In case of (either binary of weighted) *directed*[4] graphs, the matrices **A** and **W** are in general not symmetric, and each node admits an *in-degree* $k_i^{in}(\mathbf{A}) = \sum_{j\neq i} a_{ji}$, an *out-degree* $k_i^{out}(\mathbf{A}) = \sum_{j\neq i} a_{ij}$, an *in-strength* $s_i^{in}(\mathbf{W}) = \sum_{j\neq i} w_{ji}$ and an *out-strength* $s_i^{out}(\mathbf{W}) = \sum_{j\neq i} w_{ij}$. Correspondingly, we can introduce the *in-degree sequence* $\mathbf{k}^{in}(\mathbf{A}) = \{k_i^{in}(\mathbf{A})\}_{i=1}^N$, the *out-degree sequence* $\mathbf{k}^{out}(\mathbf{A}) = \{k_i^{out}(\mathbf{A})\}_{i=1}^N$, the *in-strength sequence* $\mathbf{s}^{in}(\mathbf{W}) = \{s_i^{in}(\mathbf{W})\}_{i=1}^N$ and the *out-strength sequence* $\mathbf{s}^{out}(\mathbf{W}) = \{s_i^{out}(\mathbf{W})\}_{i=1}^N$.

The degree(s) and strength(s) defined above are in some sense the immediate, first-order structural properties that can be measured in any network. For these reason, we will refer to the degree and strength sequences as the *local topological properties* of a network. To speak in general terms more easily, we will denote a generic sequence of such local constraints with the vector $\mathbf{C}(\mathbf{G})$, where $\mathbf{G}$ denotes a generic graph (either binary or weighted, either directed or undirected) and $\mathbf{C}$ denotes a generic sequence of constraints (e.g. $\mathbf{k}$ or $\mathbf{s}$) or a concatenation of more sequences (e.g. the concatenation of $\mathbf{k}^{out}$ and $\mathbf{k}^{in}$, or of $\mathbf{s}^{out}$ and $\mathbf{s}^{in}$).

The importance of local topological properties comes from the fact that, in most situations, they directly reflect the effects of ‘size’ or ‘importance’ of nodes. For instance, more popular people naturally have a higher degree in a social network, and more wealthy companies or countries naturally have a higher strength in an economic network. Clearly, one expects the size and/or importance of nodes to have a strong impact on the realized patterns of connections. For various reasons, one would like to characterize this effect quantitatively by constructing (ensembles of) networks that have the same local properties of a given real-world network. For instance, if one has empirical access only to the degrees and/or strengths of nodes of a network, then the best guess one can make about the unknown network is given by a suitable ensemble of graphs matching the empirical local properties. This is the problem of network reconstruction that will be treated extensively in Chap. 4. Another example is encountered when looking for higher-order patterns in a real network, i.e. for topological features that *cannot* be explained or replicated starting from the knowledge of only the local properties. In this case, which is the problem of pattern detection that will be treated extensively in Chap. 3, one requires a benchmark model constructed from only the local properties themselves. Both challenges require the introduction of ensembles of networks with given local properties.

Having clarified the importance of constructing graph ensembles tailored on the empirical values of the degrees and/or strengths of nodes, one might at this point wonder whether such values may be produced as random fluctuations around a common average value (in which case the model would only require the average value as a parameter, besides a choice of the probability distribution of the random fluctuations

[4] A *directed* graph is a graph where a direction is specified for each edge (self-loops are not allowed in this case as well). A directed graph is binary (or simple) if any two nodes i and j are connected in one of the following four mutually-exclusive ways: via only a directed link from i to j, via only a directed link from j to i, via both such links, or via no link at all. A directed graph is weighted if links can carry different intensities, including when they are pointing in opposite direction between the same two nodes. Again, we will assume non-negative integer weights.

around it) or whether more complicated and higher-dimensional models, controlling the local constraints for each node separately, are needed. The answer to this question has been given over decades of extensive empirical analyses which have conclusively shown that the empirical values of the degrees and the strengths observed in most real-world networks are in some sense 'irreducible' to the outcome of any simple homogeneous model. For instance, in most real-world networks both the empirical *degree distribution*[5] and the empirical *strength distribution*[6] turn out to be very broad, and typically with a right tail decaying as a *power law* of the form $P(x) \propto x^{-\gamma}$, with $2 < \gamma < 3$. In the abstract limit where the number of nodes becomes infinite, the variance of these distributions diverges while the mean remains finite, implying that the average value is not representative of the value of individual nodes. This signals the absence of a typical scale for the degree or strength of nodes. For this reason, most empirical networks are called *scale-free* [24]. The degree and strength distribution of these networks is much broader than would be obtained under a simple homogeneous network formation model with just a global constraint on e.g. the average degree or the average strength of nodes, even after including noise or stochasticity.

For instance, the oldest and most popular random graph model, the Erdős-Rényi (ER) model [25], constructs a simple binary random graph with N nodes by connecting each (distinct) pair of these nodes with a given probability p. Since each node has $N-1$ potential other nodes to connect to, and since the same value p of the probability is used for all pairs of nodes, it immediately follows that the expected[7] degree of each node i has the same value $\langle k_i \rangle = p(N-1)$. This is already an indication of the complete homogeneity of the ER model. Moreover, it is easy to show that, for each node i, the probability for the degree k_i taking a particular value k is distributed binomially in k around the above expected value $p(N-1)$. Since a binomial distribution is much narrower than typical empirical degree distributions, it is intuitively clear that the latter cannot be regarded as typical realizations of the ER model. This argument can be confirmed in various statistically rigorous ways, although we will not focus on this issue in this book. Note that the parameter p has a direct control on the expected total number of links $\langle L \rangle = pN(N-1)/2$, where $N(N-1)/2$ is the number of pairs of N nodes (i.e. the maximum possible number of edges). Therefore one can regard the ER model as an ensemble of random graphs with a global constraint on the expected total number $\langle L \rangle$ of links, or equivalently on the expected average degree[8] $\langle \bar{k} \rangle = 2\langle L \rangle / N = p(N-1)$. It is then clear that such a global, overall constraint would not produce realistic network configurations. This

[5] The *empirical degree distribution* is defined, for a given network, as the fraction $P(k)$ of nodes that have degree k.

[6] The *empirical strength distribution* is defined, for a given network, as the fraction $P(s)$ of nodes that have strength s.

[7] Throughout the book, by *expected value* (or *expectation*) of a topological property we mean the average of that property over the ensemble of random graphs under consideration. We denote expectation values with angular brackets $\langle \cdot \rangle$. The rigorous definition is given later in Eq. (2.7).

[8] The *average degree* in a simple undirected graph with N nodes is defined as $\bar{k} = N^{-1} \sum_{i=1}^{N} k_i$ and necessarily equals $2L/N$, where L is the total number of links.

calls for more complicated models where the (expected) degree of each node can be controlled independently of the degree of the other nodes.

An almost identical argument holds for the strengths. One can define the *weighted random graph model* (WRG) [19] as the weighted counterpart of the ER model, where the only constraint is now the expected total weight $\langle W \rangle$ of all links in the network, or equivalently the expected average strength[9] $\langle \bar{s} \rangle = 2\langle W \rangle / N$. It can be shown that this constraint can be implemented by going over all pairs of nodes and placing an edge of weight w according to a geometric probability distribution having the same parameter value for all node pairs. The resulting strength of all nodes is distributed according to a negative binomial distribution with the same expected value. Empirical strength distributions are therefore incompatible with typical realisations of the WRG. More complicated models of weighted networks, with separately controllable strenghts, are needed in order to restore compatibility with the heterogeneity of real-world networks.

The above discussion clarifies that, in order to construct ensembles of constrained networks that are both practically useful and theoretically sound, one should introduce a way of controlling each local property (i.e. each degree and/or strength) separately. It is useful at this point to look back at Fig. 2.1. We denote the particular initial graph (step 1) by $\mathbf{G}^*$ and the corresponding numerical value of the constraints (step 2) by $\mathbf{C}^* \equiv \mathbf{C}(\mathbf{G}^*)$. The third step will generate a collection of many graphs $\{\mathbf{G}\}$ which include $\mathbf{G}^*$ itself. It should be noted that the constraints $\mathbf{C}$ define the *sufficient statistics* of the problem: the construction of the ensemble should be possible by knowing only the value $\mathbf{C}^*$ (i.e. skipping step 1) and no other property of the graph $\mathbf{G}^*$. While this idea is conceptually simple, implementing it correctly is very challenging. Understanding the origins of this difficulty is a key step towards the appreciation of the maximum-entropy method that will be described in Sect. 2.2. For this reason, in the rest of this section we briefly review the problem by discussing various alternative attempts at the construction of ensembles of graphs with local constraints.

2.1.2 Computational Approaches

For concreteness, let us consider the case of binary undirected graphs, which is by far the most frequently explored situation. We will consider many other ensembles later in the book. The ensemble of binary undirected graphs with specified degree sequence $\mathbf{C}^* \equiv \mathbf{k}^*$ is known as the *binary configuration model* (BCM) [1, 2, 23].

Given a real-world binary undirected network $\mathbf{G}^* \equiv \mathbf{A}^*$, an entirely 'bottom-up' computational approach to the generation of the associated binary configuration model with degree sequence $\mathbf{k}^* \equiv \mathbf{k}(\mathbf{A}^*)$ consists in initially assigning each vertex i a number of 'edge stubs' equal to the target degree k_i^*. Then, pairs of stubs are

[9] The *average strength* in a weighted undirected graph with N nodes is defined as $\bar{s} = N^{-1} \sum_{i=1}^{N} s_i$ and necessarily equals $2W/N$, where W is the total weight of all links in the network.

randomly matched avoiding the formation of self-loops and multiple links, until all degrees reach their desired values (*edge stub connection*). Looking back at Fig. 2.1, this implementation has the desirable property that one can start from 'step 2' in the ensemble construction. Indeed, the edge stubs are precisely the half-edges portrayed inside the second box in the picture. Unfortunately, if the values of the degrees are too heterogeneous, this procedure is known to get stuck in configurations where vertices requiring additional connections have no more eligible partners [1, 2]. Typical realizations of the procedure share this problem, which therefore cannot be easily circumvented by simply aborting the unsuccessful realizations and starting over again.

A popular alternative method is based on a 'top-down' implementation where the entire real network $\mathbf{A}^*$ is taken as the initial configuration, and a family of randomized variants is generated by iteratively applying a *local rewiring algorithm* (LRA). In the LRA, two edges (A, B) and (C, D) are randomly selected and replaced by the two edges (A, D) and (C, B), if the latter are both not already present [1, 2] (see Fig. 2.2 for an illustration). Technically, the above procedure generates an ensemble where all randomized networks have exactly the same degree sequence as the original network. This method has been applied to various networks, including the Internet [2], cellular networks [3] and food webs [8], in order to detect higher-order patterns (such as clustering and motifs) not merely due to local constraints. Unfortunately, this approach is time-consuming since many (a number R much larger than the observed number of links L [1, 20], even if not rigorously specified) iterations of the LRA are required to obtain a *single* randomized network, and the entire process must be repeated several times to produce a large number M (again unspecified) of randomized networks, on each of which any topological property X of interest must be measured explicitly and averaged at the end to obtain an estimate for $\langle X\rangle$. The computational time required to obtain $\langle X\rangle$ is therefore of the order $O(M \cdot T_R \cdot R) + O(M \cdot T_X)$, where T_R is the average time required to perform a single successful rewiring step and T_X is that required to compute X on a single network in the randomized set. Moreover, even if the sufficient statistics of the problem is the degree sequence $\mathbf{k}(\mathbf{A}^*)$ alone, the above approach requires the entire original network $\mathbf{A}^*$ (or any other network with the same degree sequence, which is however difficult to obtain from scratch due to the problems discussed above) as the starting configuration, thus making use of much more information than required in principle.

Besides these practical limitations, the main problem of the LRA is the fact that it is *biased*, i.e. it does not sample the desired ensemble uniformly. This has been rigorously shown relatively recently [21, 26, 27]. For undirected networks, uniformity has been shown to hold, at least approximately, only when the degree sequence is such that [27]

$$k_{max} \cdot \overline{k^2}/(\overline{k})^2 \ll N \tag{2.1}$$

where k_{max} is the largest degree in the network, $\overline{k}$ is the average degree, $\overline{k^2}$ is the second moment, and N is the number of vertices. Clearly, the above condition sets an upper bound for the heterogeneity of the degrees of vertices, and is violated if the heterogeneity is strong. This is another indication that the available methods break

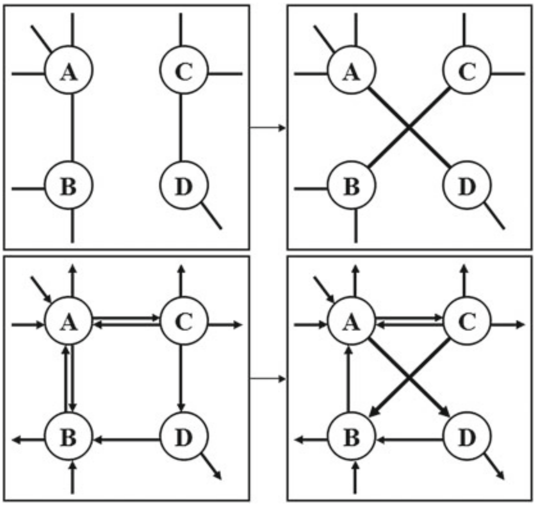

Fig. 2.2 An illustration of the local rewiring algorithm whose iteration allows to computationally explore the configuration model with sharp constraints (upper panel, for undirected networks; lower panel, for directed networks)

down for 'strongly heterogeneous' networks. As we discuss later, most real-world networks are found to fall precisely within this class.

For directed networks, where links are oriented and the constraints to be met are the numbers of incoming and outgoing links (in-degree and out-degree) separately, a condition similar to Eq. (2.1) holds, but there is also the additional problem that the LRA is non-ergodic, i.e. it is in general not able to explore the entire ensemble of networks [26]. The violation of uniformity and ergodicity in the LRA implies that the average quantities over the graphs it generates are biased, i.e. they do not correspond to the correct expectations.

It has been shown that, in order to restore ergodicity, it is enough to introduce an additional 'triangular move' inverting the direction of closed loops of three vertices [26]. However, in order to restore uniformity, one must do something much more complicated: at each iteration, the attempted 'rewiring move' must be accepted with a probability that depends on some complicated property of the current network configuration [21, 26, 27]. Since this property must be recalculated at each step, the resulting algorithm is extremely time consuming.

Other recent alternatives [28–30] rely on theorems, such as the Erdös-Gallai [31] one, that set necessary and sufficient conditions for a degree sequence to be *graphic*, i.e. realized by at least one graph. These 'graphic' methods exploit such (or related) conditions to define biased sampling algorithms in conjunction with the estimation of the corresponding sampling probabilities, thus allowing one to statistically reweight the outcome and sample the ensemble effectively uniformly [28–30]. Del Genio et al. [28] show that, for networks with power-law degree distribution of the form $P(k) \sim k^{-\gamma}$, the computational complexity of sampling *one* graph using their algorithm is $O(N^2)$ if $\gamma > 3$. However, when $\gamma < 3$ the computational complexity increases to $O(N^{2.5})$ if

$$k_{max} < \sqrt{N} \tag{2.2}$$

and to $O(N^3)$ if $k_{max} > \sqrt{N}$. The upper bound $\sqrt{N}$ is a particular case of the so-called 'structural cut-off' that we will discuss in more detail later. For the moment, it enough for us to note that Eq. (2.2) is another indication that, for strongly heterogeneous networks, the problem of sampling gets more complicated. As we will discuss later, most real networks violate Eq. (2.2) strongly.

So, while 'graphic' algorithms do provide a solution for every network, their complexity increases for networks of increasing (and realistic) heterogeneity. A more fundamental limitation is that they can only handle the problem of binary graphs with given degree sequence. The generalization of these methods to other types of networks and other constraints is not straightforward, as it would require the proof of more general 'graphicality' theorems, and *ad hoc* modifications of the algorithm.

For what concerns weighted networks, the available 'hard' algorithms regard each link weight as an integer multiple w of a fundamental unit of weight, transform each edge of weight w into w edges of unit weight and rewire the latter as in the unweighted case, now ensuring that the strength of each vertex is preserved. This means replacing a list of $L^* \leq N(N-1)/2$ weighted links, summing up to a total weight $W^* = \sum_{i<j} w^*_{ij}$, with $W^* \gg N(N-1)/2$ unweighed links. As real networks have broadly distributed weights summing up to a large W^*, this procedure becomes very time consuming as unfeasibly many rewiring steps per randomized variant must be performed. Moreover, much less is known about the potential bias produced by this algorithm in the case of weighted networks.

2.1.3 Analytical Approaches

In contrast with computational methods, analytical approaches seek to provide explicit mathematical expressions that directly estimate the ensemble averages of topological properties, without generating the ensemble computationally. Two main approaches exist. One makes use of generating functions for the relevant probability distributions [23]. For the binary configuration model, the key quantity is the generating function $g(z) = \sum_k z^k P(k)$ of the degree distribution. Unfortunately, this method assumes the network to be infinite and locally tree-like (even if in some cases this approximation turns out to perform unexpectedly well even beyond its formal range of applicability [32]), and is thus in general inappropriate if the size of the network is small and if the input degree distribution can only be realized by dense and/or clustered networks. In this approach, clustered or dense networks can only be generated by imposing additional constraints besides the degree sequence, such as the number of triangles attached to vertices [33], thus leading to a different ensemble which is not the one we are seeking to characterize. A different approach looks for an analytical expression for the probability p_{ij} that the vertices i and j are connected in the randomized ensemble [4]. Due to its probabilistic nature, this approach generates an ensemble with *soft* constraints, i.e. where graphs violating the constraints are present and assigned non-zero probabilities. The constraints are still realized on average, i.e. the expectation value $\langle \mathbf{C} \rangle$ of $\mathbf{C}$ is still equal to $\mathbf{C}^*$. The popular expression used for p_{ij} is

$$p_{ij} = \frac{k_i^* k_j^*}{2L^*} \tag{2.3}$$

where $L^* \equiv L(\mathbf{A}^*) = \sum_i k_i(\mathbf{A}^*)/2 = \sum_{i<j} a_{ij}^*$ is the total number of links. While the expected degree $\langle k_i \rangle = \sum_j p_{ij}$ generated by the above formula coincides (approximately, as we discuss below) with the desired degree k_i^*, the probability p_{ij} may exceed 1 for pairs of highly connected nodes such that $k_i^* k_j^* > 2L^*$. In general, only if the degree sequence is such that

$$k_i^* < \sqrt{2L^*} = \sqrt{\sum_j k_j^*} \quad \forall i \tag{2.4}$$

then using Eq. (2.3) on the real network $\mathbf{A}^*$ will not lead to the above problem. While the above condition is typically obeyed by networks with narrow degree distribution it is generally violated by scale-free networks displaying a power-law degree distribution $P(k) \sim k^{-\gamma}$, and this violation becomes stronger and stronger as the density of the network increases. In particular, it is easy to see that in order to ensure Eq. (2.4) the maximum degree $k_{max}^* = \max_i k_i^*$ in the network should not exceed the so-called *structural cut-off* $k_c \sim N^{1/2}$ [34]. This is particularly evident for sparse networks where the average degree $\bar{k} = \sum_i k_i/N = 2L/N$ remains constant as N increases, so that Eq. (2.4) remains valid only if $k_{max} < \sqrt{2L} \sim N^{1/2}$. By contrast, extreme value theory shows that in networks with degree distribution $P(k) \sim k^{-\gamma}$ the maximum degree scales as $k_{max}^* \sim N^{1/(\gamma-1)}$, so that if $\gamma < 3$ (as observed in most real-world scale-free networks) then $k_{max}^* > N^{1/2}$ which exceeds k_c.

Loosely speaking, the meaning of p_{ij} being larger than 1 for some pairs of vertices in Eq. (2.3) is that i and j should be connected by more than one undirected edge in order to actually realize the desired degree sequence. Also, since the desired equality $\langle k_i \rangle = k_i^*$ is only ensured if one lets the sum in $\sum_j p_{ij} = \langle k_i \rangle$ run over all vertices *including i itself*, one must allow the presence of self-loops in the randomized networks. Thus, even if this is not evident at a first glance, the ensemble generated by Eq. (2.3) does not only contain binary and loop-less undirected graphs and is thus not a proper null model for an empirical binary loop-less network $\mathbf{A}^*$ with degree sequence $\mathbf{k}^*$ violating Eq. (2.4), as is typically the case for real-world networks with broad degree distributions.

An elegant proof that the correct ensemble probability p_{ij} for loop-less graphs with no multiple connections differs from Eq. (2.3) has been proposed [5] and re-derived within the framework of maximum-entropy graph ensembles [10]. An independent proof of the inadequacy of Eq. (2.3) is that it does not generate the graph $\mathbf{A}^*$ with maximum likelihood [35]. These results show that the functional form of p_{ij} in Eq. (2.3) is intrinsically problematic and does not give highest likelihood to $\mathbf{A}^*$ and to all other graphs with the same degree sequence as $\mathbf{k}^*$.

We can briefly make a similar comment for weighted networks with given strength sequence $\mathbf{s}^*$, an ensemble known as *weighted configuration model* [11] (and discussed at length in Sect. 2.2.3 and Chaps. 3 and 4). A (naïve, yet widely used) generalization [11, 36] of the (naïve, yet widely used) expression (2.3) states that the expected weight of the link between nodes i and j in this ensemble is

$$\langle w_{ij} \rangle = \frac{s_i^* s_j^*}{2W_{tot}^*} \tag{2.5}$$

where $W_{tot}^* \equiv W_{tot}(\mathbf{W}^*) = \sum_i s_i(\mathbf{W}^*)/2 = \sum_{i<j} w_{ij}^*$ is the total weight. The above expression has been shown to have as many limitations as its binary counterpart, and to be incorrect [18]. A simple signature of this inadequacy is the fact that, although Eq. (2.5) is treated as an expected value, there is no indication of the probability distribution from which it is derived. Therefore, it is impossible to derive the expected value of topological properties that are nonlinear functions of the weights.

Therefore, while the available analytical methods are useful to characterise artificially generated networks with special properties, they cannot be used to correctly describe ensembles of networks that are realistically small, clustered, or dense. Unfortunately, the above limitations are generally ignored, and Eqs. (2.3) and (2.5) are frequently used beyond their limits of applicability to estimate connection probabilities and expected link weights. Analogous problems exist in the case of directed networks.

2.2 The Maximum-Entropy Method

The discussion in the previous section highlights that none of the above implementations succeeds in obtaining the properties of ensembles of constrained networks such that two requests are met simultaneously:

- the method is general and works for any network, even if displaying small size, high density and large clustering;
- expected values across the ensemble are unbiased and can be computed analytically, without sampling the configuration space explicitly.

In this section, we introduce a different method that fulfills the above criteria. The method is based on the maximum-entropy principle and leads to exact expressions for the probability of occurrence of any graph. It therefore allows us to calculate, correctly and analytically, the expected topological properties of graphs in the ensemble. We first illustrate the methodology in full generality, i.e. by considering an abstract choice of topological constraints, and then work out two explicit examples in detail. More examples will be given throughout the rest of the book when needed to address specific problems.

Looking again at Fig. 2.1, let us denote by $\mathbf{G}$ a generic network in the ensemble, and by $\mathbf{G}^*$ the particular original network (we may think of it as the empirical network we need to randomize). The chosen constraint is $\mathbf{C}^* = \mathbf{C}(\mathbf{G}^*)$. The ensemble consists of all possible networks $\{\mathbf{G}\}$ with the same number N of nodes and of the same type (undirected/directed, binary/weighted, etc.) as $\mathbf{G}^*$, and includes $\mathbf{G}^*$ itself. Note that

$\mathbf{G}$ can always be thought of as a matrix with entries $\{g_{ij}\}$, where g_{ij} represents the (either binary or non-negative) weight of the edge (i, j). Any topological property X evaluates to $X(\mathbf{G})$ when measured on the particular network $\mathbf{G}$, i.e. it is an (arbitrarily complicated) function of the entries $\{g_{ij}\}$.

Each graph $\mathbf{G}$ in the ensemble has an occurrence probability $P(\mathbf{G})$ whose form is determined by the particular constraints enforced. This probability must always be such that

$$\sum_{\mathbf{G}} P(\mathbf{G}) = 1 \tag{2.6}$$

where the sum runs over all graphs in the ensemble. The expectation or mean value of any topological property X is the ensemble average

$$\langle X \rangle \equiv \sum_{\mathbf{G}} X(\mathbf{G}) P(\mathbf{G}). \tag{2.7}$$

At this point, we look for the probability distribution that maximizes the Shannon-Gibbs entropy

$$S(P) \equiv -\sum_{\mathbf{G}} P(\mathbf{G}) \ln P(\mathbf{G}) \tag{2.8}$$

subject to the normalization condition (2.6) and to the desired constraints $\mathbf{C}^*$. The entropy $S(P)$ is a measure of the level of uncertainty, or randomness, in the outcome of the random variable described by the probability distribution P. Variables that have a certain outcome, i.e. whose probability is one for such outcome and zero for all other outcomes, correspond to zero entropy. On the contrary, variables that are maximally uncertain, i.e. for which every possible outcome has exactly the same probability, yield the maximum value[10] of the entropy. Maximizing the entropy subject to constraints is widely used in statistical mechanics and information theory, and in general for problems with incomplete information [37–40]. The deep meaning of constrained entropy maximization is that, in absence of any information other than the knowledge of $\mathbf{C}^*$, the probability should make the outcome of the random variable ($\mathbf{C}$ in this case) maximally uncertain provided that the constraints are met. Otherwise, the probability would be favouring specific configurations, making them more predictable than others and introducing an unjustified bias. Now, the solution to the entropy maximization problem depends on whether we want the constraints $\mathbf{C}^*$ to be *hard* or *soft*.

Enforcing hard constraints means that we only allow (i.e. assign non-zero probability) the graphs that match the constraints *exactly*, i.e. such that $\mathbf{C}(\mathbf{G}) = \mathbf{C}(\mathbf{G}^*)$. This means that, in the above definition of entropy, we can restrict the sum to such configurations only. It is easy to see that the resulting maximum-entropy distribution, which is known as the *microcanonical ensemble* in statistical physics, is uniform over

[10] The maximum value of the entropy $S(P)$ depends on the total number of configurations over which the sum in Eq. (2.8) runs. This number can be rescaled to one for all probability distributions, upon normalizing $S(P)$ by the maximum value itself.

the set of graphs that match the hard constraints:

$$P_{\text{mic}}(\mathbf{G}) = \begin{cases} 1/\Omega_{\mathbf{C}^*} & \text{if } \mathbf{C}(\mathbf{G}) = \mathbf{C}(\mathbf{G}^*) \\ 0 & \text{otherwise} \end{cases} \tag{2.9}$$

where $\Omega_{\mathbf{C}^*}$ denotes the number of graphs for which $\mathbf{C}(\mathbf{G}) = \mathbf{C}^*$. An intuitive picture of microcanonical ensembles of graphs is given in Fig. 2.3. Since $\Omega_{\mathbf{C}^*}$ is a combinatorial quantity, the above result establishes an important connection between statistical physics, probability theory and combinatorics. This connection will be explored in detail in Chap. 5. At this point, one should note that, while for simple constraints (such as the total number of links) it is easy to compute $\Omega_{\mathbf{C}^*}$, for more complicated constraints (including the degree sequence and the other local constraints we are interested in this book) this can become a very hard task. For instance, enumerating the number of graphs with a given degree sequence $\mathbf{k}^*$ is an open problem, and asymptotic expressions are known only in some restricted regime of density of the graph, i.e. under certain conditions that $\mathbf{k}^*$ must obey. For this reason, microcanonical graph ensembles are hard to deal with analytically and they are most often sampled computationally using the techniques we described in Sect. 2.1.2. However, as we discussed, these techniques are either computationally unfeasible or affected by the problem of bias, i.e. they do not sample the space of graphs according to the correct uniform probability (2.9). The computational difficulties are therefore related to the difficulties in calculating $\Omega_{\mathbf{C}^*}$ explicitly.

On the other hand, enforcing *soft* constraints means requiring that the desired value $\mathbf{C}^*$ is met only *on average* over the ensemble, or in other words that the constraint is $\langle \mathbf{C} \rangle = \mathbf{C}^*$. This requirement defines what is known as the *canonical ensemble* in statistical physics. However, unlike the traditional examples in physics, where the total energy is the only (scalar) constraint, for the cases of interest here the number of constraints grows linearly with the number of nodes in the system, since $\mathbf{C}$ is a vector of node-specific quantities. This important difference has enormous consequences, as we will discuss in Chap. 5. The form of the probability P_{can} in the canonical ensemble is found by requiring that, in addition to Eq. (2.6), the constraints are given by

$$\langle \mathbf{C} \rangle = \sum_{\mathbf{G}} \mathbf{C}(\mathbf{G}) P_{\text{can}}(\mathbf{G}) = \mathbf{C}^*. \tag{2.10}$$

It is easy to show [10] that the corresponding solution to the constrained entropy maximization problem is found by introducing a vector of Lagrange multipliers $\boldsymbol{\theta}$, one for each of the constraints in $\mathbf{C}$. The resulting conditional (on the value of $\boldsymbol{\theta}$) probability reads

$$P_{\text{can}}(\mathbf{G}|\boldsymbol{\theta}) = \frac{e^{-H(\mathbf{G},\boldsymbol{\theta})}}{Z(\boldsymbol{\theta})} \tag{2.11}$$

where $H(\mathbf{G}, \boldsymbol{\theta})$ is the so-called *graph Hamiltonian* defined as the linear combination

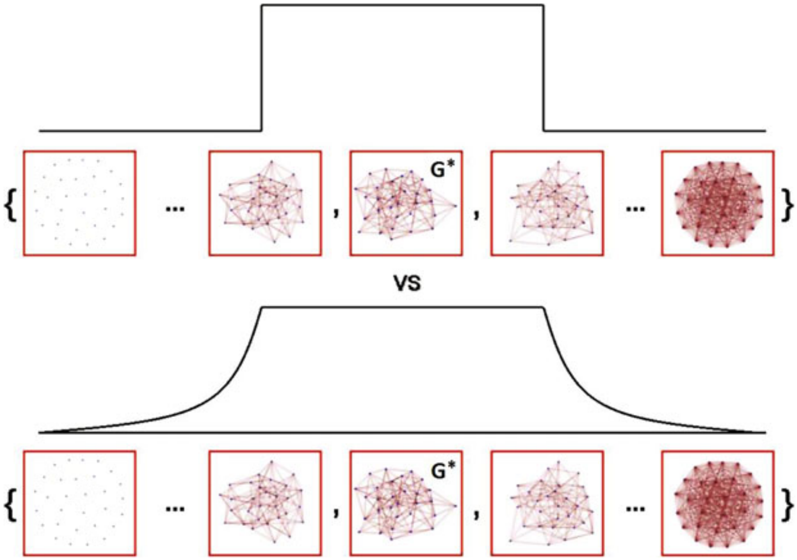

Fig. 2.3 Difference between microcanonical and canonical ensembles. Top: the microcanonical probability P_{mic} is non-zero only for the subset of graphs that realize the enforced constraints $\mathbf{C}^*$ exactly. Bottom: by contrast, the canonical probability P_{can} is non-zero for all graphs with the prescribed number of nodes, including those that violate the constraints (thus ranging from the empty graph to the complete graph), and has a constant value P_{can} for all graphs for which P_{mic} is non-zero. In general, P_{can} has the same value for all graphs that have the same value of $\mathbf{C}$

$$H(\mathbf{G}, \boldsymbol{\theta}) \equiv \sum_a \theta_a C_a(\mathbf{G}) = \boldsymbol{\theta} \cdot \mathbf{C}(\mathbf{G}) \tag{2.12}$$

and the normalizing quantity $Z(\boldsymbol{\theta})$ is the so-called *partition function*, defined as

$$Z(\boldsymbol{\theta}) \equiv \sum_{\mathbf{G}} e^{-H(\mathbf{G}, \boldsymbol{\theta})}. \tag{2.13}$$

The above results show that the graph probability $P_{\mathrm{can}}(\mathbf{G}|\boldsymbol{\theta})$ always depends on the value $\boldsymbol{\theta}$, which in turn depends on the constraints considered. As a consequence, we can rewrite Eq. (2.7) more explicitly as a function of $\boldsymbol{\theta}$:

$$\langle X \rangle_{\boldsymbol{\theta}} \equiv \sum_{\mathbf{G}} X(\mathbf{G}) P_{\mathrm{can}}(\mathbf{G}|\boldsymbol{\theta}) \tag{2.14}$$

where $\langle \cdot \rangle_{\boldsymbol{\theta}}$ denotes that the ensemble average is evaluated at the particular parameter choice $\boldsymbol{\theta}$. The above expression clarifies that the expectation value of any topological property X depends on the specific enforced constraints through $\boldsymbol{\theta}$. Different choices of the constraints imply different values of $\boldsymbol{\theta}$, $P(\mathbf{G}|\boldsymbol{\theta})$ and $\langle X \rangle_{\boldsymbol{\theta}}$. Importantly, $P_{\mathrm{can}}(\mathbf{G}|\boldsymbol{\theta})$ depends on $\mathbf{G}$ only through $\mathbf{C}(\mathbf{G})$. This automatically implies that the canonical ensemble is unbiased, i.e. graphs with the same value of the constraints are assigned equal probability. A pictorial representation of this property is given in Fig. 2.3.

Now, a crucial difference between the microcanonical and canonical ensembles is that, if $\mathbf{C}^*$ is a vector of local topological constraints, P_{mic} cannot be exactly factorized into probabilities that involve distinct pairs of nodes, whereas P_{can} can. This implies that the exact computation of $\boldsymbol{\theta}^*$ is feasible even if that of $\Omega_{\mathbf{C}^*}$ is not. For these reasons, which will be illustrated in explicit examples later on, the canonical ensemble offers a viable and exact solution to the problem of constructing ensembles of graphs with local constraints. It will be the main tool we will use throughout the rest of the book. In statistical physics, results obtained within the canonical ensemble are generally expected to become equivalent to those that would be obtained within the microcanonical one in the so-called *thermodynamic limit*.[11] This notion is called *ensemble equivalence*. Whether or not ensemble equivalence holds also in the case of local constraints is an intriguing question, and its answer is postponed to Chap. 5. Here and in Chaps. 3 and 4, we assume that enforcing the constraint $\mathbf{C}^*$ softly is a perfectly acceptable strategy, for instance because its measured value may have been corrupted by noise or error, and we are therefore inclined to accept other values around $\mathbf{C}^*$ in the ensemble construction.

2.2.1 Maximum-Likelihood Parameter Estimation

Maximum-entropy graph ensembles generated by Eq. (2.11) have been used extensively to characterize mathematically networks with specified properties [5, 7, 10, 17, 18]. However, traditionally the Lagrange multipliers $\{\theta_a\}$ have been considered as free parameters, generally drawn from carefully chosen probability densities [10, 17, 18] that allow for analytical results, in terms of which the properties of the network model have been investigated. In most cases, the aim has been to explore the topological properties in the thermodynamic limit $N \to \infty$, where N is the number of vertices of the network. This means that only generic statistical properties of real networks, such as a power-law degree distribution with a certain exponent, have been used to generate the ensemble. However, this implies that the specific properties of a particular real network (such as deviations of individual vertices from the fitted degree distribution, the intrinsic finiteness of the system, etc.) have been ignored and, more importantly, that it has not been possible to establish any correspondence between the vertices of the real network and those of the model. Thus these approaches have not allowed maximum-entropy graph ensembles to be considered as null models of *a particular real network* in order to detect empirical topological patterns, or to

[11] In statistical physics, the *thermodynamic limit* is defined as the limit where the number of fundamental units that describe the microscopic configurations of the system diverges. In our graph ensembles, we regard the nodes as the units and their connections as the interactions.

reconstruct network topology from partial information, or even to enumerate graphs compatible with a specified vector of constraints.

Now, following [41], we make a step forward and construct, for a given choice of the constraints, the particular maximum-entropy graph ensemble representing the family of correctly randomized counterparts of a given real network $\mathbf{G}^*$. Explicitly, we consider a canonical ensemble of graphs with the same number N of vertices as the real network, and for a given choice of the constraints we fit the model defined by Eq. (2.11) to the empirical network $\mathbf{G}^*$. To this end, we exploit previous results [35] showing that maximum-entropy graph ensembles defined by Eq. (2.11) are a particular class of models for which the maximum-likelihood principle provides an excellent way to estimate parameters. In particular, it can be easily shown [35] that the log-likelihood

$$\mathscr{L}(\boldsymbol{\theta}) \equiv \ln P(\mathbf{G}^*|\boldsymbol{\theta}) = -H(\mathbf{G}^*, \boldsymbol{\theta}) - \ln Z(\boldsymbol{\theta}) \tag{2.15}$$

is maximized by the particular value $\boldsymbol{\theta}^*$ such that the ensemble average $\langle C_a \rangle_{\boldsymbol{\theta}^*}$ of each constraint C_a equals the empirical value $C_a(\mathbf{G}^*)$ measured on the real network:

$$\langle \mathbf{C} \rangle^* \equiv \langle \mathbf{C} \rangle_{\boldsymbol{\theta}^*} = \sum_{\mathbf{G}} \mathbf{C}(\mathbf{G}) P(\mathbf{G}|\boldsymbol{\theta}^*) = \mathbf{C}(\mathbf{G}^*) \tag{2.16}$$

where we have used $\langle \cdot \rangle^*$ as a shorthand notation to indicate the ensemble average $\langle \cdot \rangle_{\boldsymbol{\theta}^*}$ evaluated at the particular value $\boldsymbol{\theta}^*$. The above results means that *the maximum likelihood principle indicates, for maximum-entropy graph ensembles, precisely the parameter choice that ensures that the desired constraints are met*. This is not true in general: in other network models, tuning the average values of the topological properties of interest to their empirical values requires a parameter choice which in general does not maximize the likelihood to obtain the real network [35], thus introducing a bias in the analysis [42–44].

Solving the maximum-likelihood equations only takes a computational time T_E which is much shorter than the time required to measure any topological property of typical interest. Moreover, the time required to compute the expectation value $\langle X \rangle$ of a given property X analytically (formally corresponding to an average over a huge number of randomized configurations) is the same as the time T_X required to compute the same property on the single original network. The artificial generation of many randomized variants of the original network is no longer required. Therefore this method takes only a total time $O(T_E + T_X)$ to obtain $\langle X \rangle$ analytically, which is incredibly shorter than the aforementioned time $O(M \cdot T_R \cdot R) + O(M \cdot T_X)$ required by the LRA to obtain $\langle X \rangle$ only approximately. Importantly, T_E is independent of the complexity of the topological property X to measure, which means that for complicated properties $O(T_E + T_X) = O(T_X)$. Therefore for any topological property X which can be measured in a large but still reasonable time $O(T_X)$ on the real network, the computation of its expectation value $\langle X \rangle$ will require the same time $O(T_X)$. If the time required in order to obtain $\langle X \rangle$ is too large, it is because the time required to measure X is too large as well. In other words, the property X is too complicated to be computed on the real network itself. In such a case, the problem is not due to the method, but to a demanding choice of X for that particular network.

Note that in Eqs. (2.14) and (2.16) the expectation values and the model parameters play inverted roles: while in Eq. (2.14) the expectation values are obtained as a function of the parameters $\boldsymbol{\theta}$ which can be varied arbitrarily, in Eq. (2.16) the observed constraints, which are measured on the particular real network and are therefore given as an input, are used to fix the model parameters to the values $\boldsymbol{\theta}^*$. Once the parameters solving the equations are found, they can be directly used to obtain the expectation value $\langle X \rangle$ and standard deviation $\sigma[X]$ of any topological property X of interest analytically (details on how to calculate standard deviations can be found in [41]). When useful, this also allows one to obtain a z-score representing the number of standard deviations by which the randomized value $\langle X \rangle$ differs form the observed value $X(\mathbf{A}^*)$. The possibility to obtain the standard deviations and z-scores is very important, because it allows one to assess which topological properties X are consistent with their randomized value $\langle X \rangle$ within a statistical error, and which deviate significantly from the null expectation. In the former case, one can conclude that the enforced constraints completely explain the higher-order property X. In the latter case, the observed property cannot be traced back to the constraints, and therefore requires additional explanations or generating mechanisms besides those required in order to explain the constraints themselves. We will discuss this procedure in more detail in the next chapter.

2.2.2 A First Worked-Out Example: Binary, Undirected Networks with Constrained Degree Sequence

In the binary, undirected case, each graph $\mathbf{G}$ is completely specified by its (symmetric) adjacency matrix $\mathbf{A}$. An important ensemble of binary undirected graphs is one where the constraint is the degree sequence [10]. This null model is also known as *configuration model* (CM). In our formalism this model is implemented by defining the following Hamiltonian:

$$H(\mathbf{A}) = \sum_i \theta_i k_i(\mathbf{A}) = \sum_i \sum_{j<i} (\theta_i + \theta_j) a_{ij} \tag{2.17}$$

and one can show [10] that this allows one to write the partition function as

$$Z(\boldsymbol{\theta}) \equiv \sum_{\mathbf{A}} e^{-H(\mathbf{A},\boldsymbol{\theta})} = \prod_i \prod_{j<i} (1 + x_i x_j) \tag{2.18}$$

and the graph probability as

$$P(\mathbf{A}) = \prod_i \prod_{j<i} p_{ij}^{a_{ij}} (1 - p_{ij})^{1-a_{ij}} \tag{2.19}$$

where

$$p_{ij} = \frac{x_i x_j}{1 + x_i x_j} \tag{2.20}$$

(with $x_i \equiv e^{-\theta_i}$) is the probability that a link exists between vertices i and j in the maximum-entropy ensemble of binary undirected graphs characterized by the given degree sequence as the constraint.

The maximum-likelihood condition [41] prescribes to find the solution $\{x_i^*\}_{i=1}^N$ to the equations

$$\langle k_i \rangle = \sum_{j \neq i} \frac{x_i x_j}{1 + x_i x_j} = k_i(\mathbf{A}^*) \quad \forall i \tag{2.21}$$

by choosing the imposed constraint to be the empirical degree sequence $\{k_i(\mathbf{A}^*)\}_{i=1}^N$ of the particular real network $\mathbf{A}^*$ or, equivalently, by finding the values of the parameters $\{x_i^*\}_{i=1}^N$ that maximize the likelihood $P(\mathbf{A}^*)$ [35, 41]. Inserting the $\{x_i^*\}_{i=1}^N$ into Eq. (2.20) allows one to easily compute the expectation value $\langle X \rangle^*$ of any topological property X analytically, without generating the randomized networks explicitly [41].

Thus, Eq. (2.20) yields the exact value of the connection probability in the ensemble of randomized networks with the same average degree sequence as the empirical one and Eq. (2.21) shows that, by construction, the degrees of all vertices are special local quantities whose expected and empirical values are exactly equal: $\langle k_i \rangle^* = k_i(\mathbf{A}^*)$. It follows that the p_{ij} coefficients can be calculated by using any of the networks in the corresponding degree sequence-constrained microcanonical ensemble.

The expectation value of any higher-order topological property can be derived exploiting the fact that $\langle a_{ij} \rangle = p_{ij}$ and that different pairs of vertices are statistically independent, which implies $\langle a_{ij} a_{kl} \rangle = p_{ij} p_{kl}$ if (i, j) and (k, l) are distinct pairs of vertices, whereas $\langle a_{ij} a_{kl} \rangle = \langle a_{ij}^2 \rangle = \langle a_{ij} \rangle = p_{ij}$ if $(i, j) = (k, l)$.

2.2.3 A Second Worked-Out Example: Weighted, Undirected Networks with Constrained Strength Sequence

In the weighted, undirected case, each graph $\mathbf{G}$ is completely specified by its (symmetric) non-negative weight matrix $\mathbf{W}$ whose entries w_{ij} will be understood as integer-valued. The ensemble with local constraints is in this case known as *weighted configuration model* (WCM) [11] and specifies the strength sequence as the constraint. The Hamiltonian therefore reads

$$H(\mathbf{W}) = \sum_i \theta_i s_i(\mathbf{W}) = \sum_i \sum_{j<i} (\theta_i + \theta_j) w_{ij} \tag{2.22}$$

and one can show that this allows to write the partition function as [10]

$$Z(\boldsymbol{\theta}) \equiv \sum_{\mathbf{W}} e^{-H(\mathbf{W},\boldsymbol{\theta})} = \prod_{i} \prod_{j<i} (1 - x_i x_j)^{-1} \tag{2.23}$$

and the graph probability as [18]

$$P(\mathbf{W}) = \prod_{i} \prod_{j<i} q_{ij}(w_{ij}) \tag{2.24}$$

where

$$q_{ij}(w) = (x_i x_j)^w (1 - x_i x_j) \tag{2.25}$$

(with $x_i \equiv e^{-\theta_i}$) is the probability that a link of weight w exists between vertices i and j in the maximum-entropy ensemble of weighted, undirected graphs, subject to specifying the given strength sequence as the constraint.

If the latter is chosen to be the empirical strength sequence $\{s_i(\mathbf{W}^*)\}$ of the particular real network $\mathbf{W}^*$, then Eq. (2.25) yields the exact value of the connection probability in the ensemble of randomized weighted networks with the same average strength sequence as the empirical one, provided that the parameters $\{x_i\}_{i=1}^N$ are set to the values $\{x_i^*\}_{i=1}^N$ that maximize the likelihood $P(\mathbf{W}^*)$ [41]. These values are the solution of the following set of N coupled nonlinear equations:

$$\langle s_i \rangle = \sum_{j \neq i} \frac{x_i x_j}{1 - x_i x_j} = s_i(\mathbf{W}^*) \quad \forall i. \tag{2.26}$$

Once the values $\{x_i^*\}_{i=1}^N$ are found, they are inserted into Eq. (2.25) which allows to easily compute the expectation value $\langle X \rangle^*$ of any topological property X analytically. Equation (2.26) shows that, by construction, the strengths of all vertices are special local quantities whose expected and empirical values are exactly equal: $\langle s_i \rangle^* = s_i(\mathbf{W}^*)$.

The expectation value of any higher-order topological property can be derived exploiting the fact that $\langle w_{ij} \rangle = \sum_w w q_{ij}(w) = x_i x_j / (1 - x_i x_j)$, and that different pairs of vertices are statistically independent, which implies $\langle w_{ij} w_{kl} \rangle = \langle w_{ij} \rangle \langle w_{kl} \rangle$ if $(i - j)$ and $(k - l)$ are distinct pairs of vertices, whereas $\langle w_{ij} w_{kl} \rangle = \langle w_{ij}^2 \rangle$ if $(i - j)$ and $(k - l)$ are the same pair of vertices. The expected value of the power of the weight between vertices i and j is calculated as follows:

$$\langle w_{ij}^{\alpha} \rangle \equiv \sum_{w} w^{\alpha} q_{ij}(w) = (1 - x_i x_j) \mathrm{Li}_{-\alpha}(x_i x_j) \tag{2.27}$$

where $\mathrm{Li}_n(z)$ denotes the Polylogarithm function defined as

$$\mathrm{Li}_n(z) \equiv \sum_{l=1}^{\infty} \frac{z^l}{l^n}. \tag{2.28}$$

The adjacency matrix representing the existence of a link (irrespective of its intensity) between vertex i and vertex j is derived from the weight matrix by setting $a_{ij} = \Theta(w_{ij})$, where $\Theta(x) = 1$ if $x > 0$ and $\Theta(x) = 0$ otherwise. The probability that vertices i and j are connected, irrespective of the edge weight, is now $\langle a_{ij} \rangle = p_{ij} \equiv 1 - q_{ij}(0) = x_i x_j$. In analogy with the expectation values of products of weights, we have $\langle a_{ij} a_{kl} \rangle = p_{ij} p_{kl}$ if $(i - j)$ and $(k - l)$ are distinct pairs of vertices, whereas $\langle a_{ij} a_{kl} \rangle = \langle a_{ij}^2 \rangle = \langle a_{ij} \rangle = p_{ij}$ if $(i - j)$ and $(k - l)$ are the same pair of vertices.

2.3 Comparing Models Obtained from Different Constraints

The two worked out examples considered above will be used extensively throughout this book, together with other models. When multiple models are applied to the same set of network data, one needs a rigorous statistical procedure to compare them and choose, loosely speaking, the 'best one'. In fact, judging a model purely on the basis of its performance in reproducing the observed trends represents a naïve way of proceeding exposed to many risks, the most dangerous one being that of preferring models that overfit the data via redundant parameters that have high inter-correlations and provide spurious information on the system [45, 46]. For instance, alternative models are often compared exclusively in terms of the values of their likelihood functions evaluated in their stationary points: the higher the value, the better the model is expected to describe the considered network. However, this procedure lacks a rigorous statistical justification and does not address the parsimony of the models, e.g. the number of parameters.

On the contrary, we would like to rely on a criterion able to unambiguously identify not only the most effective null model in explaining empirical data, but also the most statistically correct one. A more appropriate way of testing the effectiveness of two competing null models (say NM_i and NM_j, where NM_j contains extra parameters with respect to NM_i) is the *Likelihood Ratio Test* (LRT) [47], which prescribes to calculate the quantity

$$D_{NM_i/NM_j} \equiv -2(\mathscr{L}_{NM_i}(\boldsymbol{\theta}_i^*) - \mathscr{L}_{NM_j}(\boldsymbol{\theta}_j^*)) \tag{2.29}$$

(where the symbols $\boldsymbol{\theta}_i$ and $\boldsymbol{\theta}_j$ indicate the two different sets of Lagrange multipliers that maximize the likelihood of the two models) and compare it to some threshold value determined by some chosen significance level. If D_{NM_i/NM_j} is smaller than the threshold, then model NM_j should be rejected even though its log-likelihood is higher than that of NM_i.

However, the LRT suffers from some limitations [47]. One lies in the fact that the competing null models have to be nested: NM_i has to be a special case of NM_j. Another limitation has to do with the number of models that can be tested: only two alternative hypoteses at a time can be compared, thus making a global ranking of all the models in our set impossible.

So, we prefer a criterion which is suitable for more than two, possibly not nested, competing null models. The *Akaike Information Criterion* (AIC) [45, 46, 48] is one such criterion. It prescribes to calculate the quantity

$$\text{AIC}^*_{NM_i} \equiv 2K_{NM_i} - 2\mathscr{L}_{NM_i}(\boldsymbol{\theta}^*_i) \quad \forall i \tag{2.30}$$

for every null model in the set and then choose the model with the lowest value. Since the above quantity is (twice) the difference between the number of parameters K of null model NM_i and its log-likelihood evaluated in its maximum, such procedure satisfies all our requirements: it is likelihood-based, it discounts the number of model parameters and allows for a comparison among several (not necessarily nested) models.

However, whenever the number n of empirical observations becomes too small with respect to the number of parameters (a rule of thumb being $n/K_{NM_i} < 40$ [45, 46]) the modified quantity

$$\text{AICc}^*_{NM_i} \equiv \text{AIC}^*_{NM_i} + \frac{2K_{NM_i}(K_{NM_i}+1)}{n - K_{NM_i} - 1}, \tag{2.31}$$

providing an extra correction term further penalizing models with many parameters, should be used. When $n \gg K^2_{NM_i}$, AICc converges to AIC and the usual form is recovered. Notice that n has no subscript because the comparison between different null models has to be carried on the same set of observations: naturally, this holds true also for AIC and, generally speaking, for all model selection methods. More precisely, in all the cases of interest for us, our samples will be constituted by the entries of the adjacency matrix, i.e. $n = N(N-1)/2$ observations when dealing with undirected networks and $n = N(N-1)$ observations when dealing with directed networks.

Both AIC and AICc select the most effective model in explaining observations, avoiding (or, at least, strongly reducing) the risk of choosing overfitting models. However, to quantify the relative improvement brought about by the best model, the so called *Akaike weights* can be computed as follows:

$$w^{\text{AIC}}_{NM_i} \equiv \frac{e^{-\frac{\varDelta_{NM_i}}{2}}}{\sum_{r=1}^{R} e^{-\frac{\varDelta_{NM_r}}{2}}} \tag{2.32}$$

where $\varDelta_{NM_i} \equiv \text{AIC}^*_{NM_i} - \min\{\text{AIC}^*_{NM_i}\}_{i=1}^{R}$, R being the total number of considered null models. The Akaike weight of a specific model is usually interpreted as the probability that that model is, in fact, the best one. Models with $\varDelta \leq 2$

are given substantial statistical support, models with $4 \leq \Delta \leq 7$ are given less support and models with $\Delta > 10$ have essentially no support [45, 46, 48–50]. Confidence intervals can also be defined [45, 46, 48–50].

An alternative criterion to AIC, the *Bayesian Information Criterion* (BIC) [45, 46, 48–50], has also been proposed and the corresponding weights defined accordingly. The only, apparently simple but actually substantial, difference lies in the term to be discounted from the maximized likelihood:

$$\text{BIC}^*_{NM_i} \equiv K_{NM_i} \ln n - 2\mathscr{L}_{NM_i}(\boldsymbol{\theta}^*_i) \quad \forall i. \tag{2.33}$$

The first addendum does not only account for the number of parameters, K_{NM_i}, but also for the cardinality of the sample, n. Since BIC discounts the sample cardinality from the very beginning, there is no need to define a corrected Bayesian Information Criterion analogous to AICc. The *Bayesian weights* are defined analogously to the Akaike weights:

$$w^{\text{BIC}}_{NM_i} \equiv \frac{e^{-\frac{\Delta^B_{NM_i}}{2}}}{\sum_{r=1}^{R} e^{-\frac{\Delta^B_{NM_r}}{2}}} \tag{2.34}$$

where now $\Delta^B_{NM_i} \equiv \text{BIC}^*_{NM_i} - \min\{\text{BIC}^*_{NM_i}\}_{i=1}^{R}$, R being the total number of considered null models. Criteria to interpret BIC weights follow the same lines stated for AIC weights [45, 46, 48–50].

Generally speaking, because of the extra term $\ln n$, BIC is believed to be more restrictive than AIC, as the former favors models with a lower number of parameters than those favored by the latter [45]. However, which criterion performs best, and under which conditions, is still debated and other model-selection methods (such as *multimodel inference*, where some form of average over different models is performed [45]) have been proposed. In this book we will use both criteria and compare them when necessary.

References

1. S. Maslov, K. Sneppen, Specificity and stability in topology of protein networks. Science **296**, 910–913 (2002)
2. S. Maslov, K. Sneppen, A. Zaliznyak, Detection of topological patterns in complex networks: correlation profile of the Internet. Physica A **333**, 529–540 (2004)
3. R. Milo, S. Shen-Orr, S. Itzkovitz, N. Kashtan, D. Chklovskii, U. Alon, Network motifs: simple building blocks of complex networks. Science **298**, 824–827 (2002)
4. F. Chung, L. Lu, “Connected components in random graphs with given expected degree sequences”, Ann. Combin. **6**(125) (2002)
5. J. Park, M.E.J. Newman, "Origin of degree correlations in the Internet and other networks". Phys. Rev. E **68**, 026112 (2003)
6. M. Catanzaro, M. Boguná, R. Pastor-Satorras, Generation of uncorrelated random scale-free networks. Phys. Rev. E **71**, 027103 (2005)

7. D. Garlaschelli, M.I. Loffredo, Multispecies grand-canonical models for networks with reciprocity. Phys. Rev. E **73**, 015101(R) (2006)
8. D.B. Stouffer, J. Camacho, W. Jiang, L.A.N. Amaral, Evidence for the existence of a robust pattern of prey selection in food webs. Proc. R. Soc. B **274**, 1931–1940 (2007)
9. R. Guimerá, M. Sales-Pardo, L.A.N. Amaral, Classes of complex networks defined by role-to-role connectivity profiles. Nat. Phys. **3**, 63–69 (2007)
10. J. Park, M.E.J. Newman, Statistical mechanics of networks. Phys. Rev. E **70**, 066117 (2004)
11. M.A. Serrano, M. Boguná, "Weighted configuration model" AIP Conf. Proc. **776**(101) (2005)
12. M.A. Serrano, M. Boguná, R. Pastor-Satorras, Correlations in weighted networks. Phys. Rev. E **74**, 055101(R) (2006)
13. M.A. Serrano, Rich-club vs rich-multipolarization phenomena in weighted networks. Phys. Rev. E **78**, 026101 (2008)
14. A. Barrat, M. Barthelemy, R. Pastor-Satorras, A. Vespignani, The architecture of complex weighted networks. Proc. Nat. Acad. Sci. **101**, 3747–3752 (2004)
15. T. Opsahl, V. Colizza, P. Panzarasa, J.J. Ramasco, Prominence and Control: The Weighted Rich-Club Effect. Phys. Rev. Lett. **101**, 168702 (2008)
16. K. Bhattacharya, G. Mukherjee, J. Saramaki, K. Kaski, S.S. Manna, "The International Trade Network: weighted network analysis and modelling", J. Stat. Mech., P02002 (2008)
17. G. Bianconi, The entropy of network ensembles. Phys. Rev. E **79**, 036114 (2009)
18. D. Garlaschelli, M.I. Loffredo, Generalized Bose-Fermi statistics and structural correlations in weighted networks. Phys. Rev. Lett. **102**, 038701 (2009)
19. D. Garlaschelli, The weighted random graph model. New J. Phys. **11**, 073005 (2009)
20. R. Milo, N. Kashtan, S. Itzkovitz, M.E.J. Newman, U. Alon, "On the uniform generation of random graphs with prescribed degree sequences", http://arxiv.org/abs/cond-mat/0312028
21. Y. Artzy-Randrup, L. Stone, Generating uniformly distributed random networks. Phys. Rev. E **72**, 056708 (2005)
22. L. Tabourier, C. Roth, J.-P. Cointet, "Generating constrained random graphs using multiple edge switches", http://arxiv.org/abs/1012.3023
23. M.E.J. Newman, S.H. Strogatz, D.J. Watts, Random graphs with arbitrary degree distributions and their applications. Phys. Rev. E **64**, 026118 (2001)
24. G. Caldarelli, *Scale-free Networks. Complex Webs in Nature and Technology* (Oxford University Press, 2007)
25. P. Erdös, A. Rényi, On random graphs. Publicationes Mathematicae Debrecen **6**, 290–297 (1959)
26. A.C.C. Coolen, A. De Martino, A. Annibale, Constrained Markovian dynamics of random graphs. J. Stat. Phys. **136**, 1035–1067 (2009)
27. E.S. Roberts, A.C.C. Coolen, Unbiased degree-preserving randomization of directed binary networks. Phys. Rev. E **85**, 046103 (2012)
28. C.I. Del Genio, H. Kim, Z. Toroczkai, K.E. Bassler, Efficient and exact sampling of simple graphs with given arbitrary degree sequence. PLoS ONE **5**(4), e10012 (2010)
29. H. Kim, C.I. Del Genio, K.E. Bassler, Z. Toroczkai, Constructing and sampling directed graphs with given degree sequences. New J. Phys **14**(2), 023012 (2012)
30. J. Blitzstein, P. Diaconis, A sequential importance sampling algorithm for generating random graphs with prescribed degrees. Internet Mathematics **6**(4), 489–522 (2011)
31. P. Erdös, T. Gallai, "Graphs with prescribed degree of vertices", Mat. Lapok **11**(477) (1960)
32. S. Melnik, A. Hackett, M.A. Porter, P.J. Mucha, J.P. Gleeson, The unreasonable effectiveness of tree-based theory for networks with clustering. Phys. Rev. E **83**, 036112 (2011)
33. M.E.J. Newman, Random graphs with clustering. Phys. Rev. Lett. **103**, 058701 (2009)
34. M. Boguná, R. Pastor-Satorras, A. Vespignani, Cut-offs and finite size effects in scale-free networks. Eur. Phys. J. B **38**, 205–209 (2004)
35. D. Garlaschelli, M.I. Loffredo, Maximum likelihood: Extracting unbiased information from complex networks. Phys. Rev. E **78**, 015101(R) (2008)
36. M.E.J. Newman, Analysis of weighted networks. Phys. Rev. E **70**, 056131 (2004)

37. J.W. Gibbs, *Elementary principles in statistical mechanics* (Charles Scribner's Sons, New York, 1902)
38. C. Shannon, A mathematical theory of communication. Bell System Tech. Jour. **27**(379–423), 623–656 (1948)
39. E.T. Jaynes, "Information theory and statistical mechanics", Phys. Rev. **106**(620) (1957)
40. E.T. Jaynes, "On the rationale of maximum-entropy methods", Proc. IEEE **70**(939) (1982)
41. T. Squartini, D. Garlaschelli, Analytical maximum-likelihood method to detect patterns in real networks. New J. Phys. **13**, 083001 (2011)
42. P. Holland, S. Leinhardt, Sociological Methodology, ed. by D. Heise. (Jossey-Bass, San Francisco, 1975), pp. 1–45
43. S. Wasserman, K. Faust, *Social Network Analysis* (Cambridge University Press, Cambridge, 1994)
44. T.A.B. Snijders, "Markov chain Monte Carlo estimation of exponential random graph models", J. Soc. Struct. **3**(2) (2002)
45. K.P. Burnham, D.R. Anderson, *Model selection and multi-model inference: a practical information-theoretic approach* (Springer, New York, 2002)
46. J.B. Johnson, K.S. Omland, Model selection in ecology and evolution. Trends Ecol. Evol. **9**, 101–108 (2004)
47. D.R. Cox, D.V. Hinkley, *Theoretical statistics* (Chapman and Hall, Boca Raton, 1974)
48. H. Akaike, A new look at the statistical model identification. IEEE Trans. Aut. Cont. **19**, 716–723 (1974)
49. K.P. Burnham, D.R. Anderson, Multimodel inference: understanding AIC and BIC in Model Selection. Soc. Met. Res. **33**, 261–304 (2004)
50. E.J. Wagenmakers, S. Farrell, AIC model selection using Akaike weights. Psych. Bull Rev. **11**, 192–196 (2004)

Springer

http://www.springer.com/978-3-319-69436-8

Maximum-Entropy Networks
Pattern Detection, Network Reconstruction and Graph Combinatorics

Squartini, T.; Garlaschelli, D.

2017, XII, 116 p. 34 illus., 31 illus. in color., Softcover

ISBN: 978-3-319-69436-8