SAMPLING METHODS FOR RANDOM SIMPLE AND BIPARTITE GRAPHS WITH PRESCRIBED DEGREE SEQUENCES

by

Abdulkadir Çelikkanat

B.S., Mathematics, Boğaziçi University, 2014

Submitted to the Institute for Graduate Studies in
Science and Engineering in partial fulfillment of
the requirements for the degree of
Master of Science

Graduate Program in Computer Engineering

Boğaziçi University

2017

SAMPLING METHODS FOR RANDOM SIMPLE AND BIPARTITE GRAPHS WITH PRESCRIBED DEGREE SEQUENCES

APPROVED BY:

Assoc. Prof. Ali Taylan Cemgil (Thesis Supervisor)	.....
Prof. A. C. Cem Say	.....
Prof. Ş. İlker Birbil	.....

DATE OF APPROVAL: 10.07.2017

ACKNOWLEDGEMENTS

Firstly, I would like to express my gratitude to my advisor Assoc. Prof. Ali Taylan Cemgil for his guidance and support during the course of my thesis research and Master studies. I would like to also thank the other member of my thesis committee, Prof. A. C. Cem Say and Prof. Ş. İlker Birbil for their insightful comments.

I am also so obliged to Assist. Prof. Tınaz Ekim Aşıcı and Assist. Prof. Ümit Işlak for attending my thesis defense and their invaluable feedback.

I want to thank my friends, Oğuzhan Metin and Ramazan Aslan for their support and help during my undergraduate and Master studies.

Above all, I am very grateful to my family for their endless support, encouragement, and love throughout my life.

ABSTRACT

SAMPLING METHODS FOR RANDOM SIMPLE AND BIPARTITE GRAPHS WITH PRESCRIBED DEGREE SEQUENCES

Complex networks have attracted considerable attention in recent years with the increase in the studies of real systems modeled by graphs such as biological and social networks. One problem in this domain is the generation of typical instances from a collection of graphs admitting certain properties, such as the degree sequence or the clustering coefficient. In this thesis, the sampling problem is addressed for simple and bipartite graphs with a given fixed degree sequence.

A natural Markov chain method relying on the edge switching steps is introduced for simple graphs. Due to the difficulties of directly obtaining samples from the uniform distribution over the set of possible realizations of a given degree sequence, algorithms using importance sampling and sequential importance sampling techniques are investigated for simple and bipartite graphs. Here, we focus on algorithms proposed by Blitzstein and Diaconis [1] and Chen *et al.* [2]. A new uniform sampling and exact counting algorithm is proposed for simple graphs by adapting, and transforming the method suggested by Miller and Harrison [3] for bipartite graphs. Lastly, applications of the algorithms are illustrated in several examples such as hypothesis testing, network analysis and graph enumeration.

ÖZET

DERECE DİZİLİ RASTGELE BASİT VE İKİ PARÇALI ÇİZGELER İÇİN ÖRNEKLEME YÖNTEMLERİ

Karmaşık ağlar son yıllarda biyolojik ve sosyal ağlar gibi çizgelerle modellenen gerçek sistemlerdeki çalışmaların artmasıyla oldukça dikkat çekti. Bu alandaki bir problem de derece dizileri veya kümelenme katsayısı gibi belirli özellikleri sağlayan çizge topluluğundan tipik örneklerin üretilmesidir. Bu tezde, basit ve iki parçalı çizgeler için örnekleme problemi ele alınmıştır.

Kenar geçiş adımlarına dayanan doğal bir Markov zinciri yöntemi basit çizgeler için sunulmuştur. Belirli bir derece dizisinin olası gerçekleşimleri üzerine doğrudan tekdüze dağılımdan örneklerin elde edilmesinin zorluklarından dolayı, basit ve iki parçalı çizgeler için önem örnekleme ve sıralı önem örnekleme tekniklerini kullanan algoritmalar araştırılmıştır. Burada, Blitzstein ve Diaconis [1] ve Chen ve arkadaşları [2] tarafından sunulan algoritmalar üzerinde duruyoruz. İki parçalı çizgeler için Miller ve Harrison [3] tarafından önerilen bir yöntemi basit çizgeler için adapte ederek ve dönüştürerek yeni bir tekdüze örnekleme ve tam sayma algoritması önerilmiştir. Son olarak algoritmaların uygulamaları hipotez testi, ağ analizi ve grafik sayımı gibi çeşitli örneklerde gösterilmiştir.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF SYMBOLS

$Bern(\pi)$	Bernoulli distribution with parameter π
$\boldsymbol{d}$	Degree sequence
$\overline{\boldsymbol{d}}$	Counts of occurrences
$\boldsymbol{d}^*$	Conjugate vector of $\boldsymbol{d}$
$\mathbb{E}[\cdot]$	Expected value operator
$\mathcal{G}^{\boldsymbol{d}}$	Simple graphs realizing the sequence $\boldsymbol{d}$
K	Transition kernel
$\mathcal{L}(\cdot)$	Left shift operator
$\mathcal{M}_{\boldsymbol{d}}$	Edge-switching Markov chain
$\mathcal{M}^{(\boldsymbol{p},\boldsymbol{q})}$	Binary matrices realizing the margin sums $(\boldsymbol{p},\boldsymbol{q})$
$N(\boldsymbol{p},\boldsymbol{q})$	Number of $(0,1)$-matrices satisfying the margins $(\boldsymbol{p},\boldsymbol{q})$
$\mathbb{N}$	$\{0,1,2,...\}$
$\mathbb{N}_+$	$\{1,2,3,...\}$
$\mathcal{N}(v)$	The set of vertices adjacent to v
$\mathbb{R}$	$(-\infty,+\infty)$
$\bar{\mathbb{R}}$	$[-\infty,+\infty]$
$\bar{\mathbb{R}}_+$	$[0,+\infty]$
$\boldsymbol{p}$	Row sum of a binary matrix
$\mathbb{P}(\cdot)$	Probability measure
$\mathcal{P}(\mathsf{X})$	The set of all subsets of X
$\boldsymbol{q}$	Column sum of a binary matrix
X	Sample space
$\mathcal{X}$	Set of events
X	Random variable
$\delta_x(\cdot)$	Dirac measure
$\lambda(\cdot)$	Counting measure

LIST OF ACRONYMS/ABBREVIATIONS

BA	Barabási–Albert
CC	Clustering Coefficient
CP	Conditional-Posisson
ER	Erdös-Renyi
HH	Havel-Hakimi
IID	Independent and Identically Distributed
IS	Importance Sampling
LLN	Law of Large Numbers
MCMC	Markov Chain Monte Carlo
SIS	Sequential Importance Sampling
WS	Watts-Strogatz

1. INTRODUCTION

A graph is referred to as *random* if it is chosen from a certain collection of graphs according to a probability law. Random graphs have taken considerable attention from various fields from sociology to computer science to biology [4], and they are used as a model for many real-world networks to study their properties under certain constraints and assumptions. In recent years, many studies have been conducted on the generation of graphs satisfying a given degree sequence since it is one of the prominent characteristics of a graph affecting its structure. For instance, the graph in Figure 1.1 illustrates relationships between a group of freshman students. Each vertex in the graph corresponds to a student and a link between a pair of students indicates that they are in a friendly relationship.

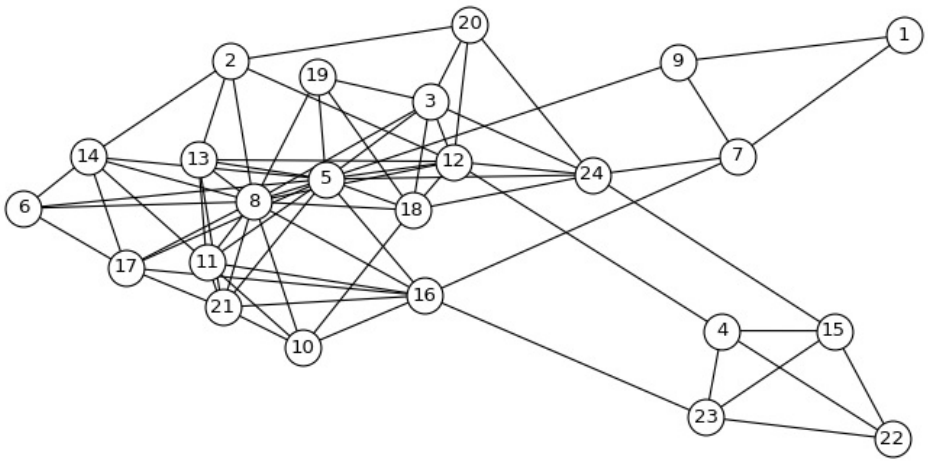


Figure 1.1. Friendship Network of Freshman Students

Although the original data [5, 6] were collected at seven different time points by asking students to evaluate their relationships with others, their assessments at the last time step is considered to construct the graph. A connection between a pair of students is placed, only if one of them considers their relationships friendly, and the degree sequence of this particular network is

$$(2, 5, 7, 4, 14, 4, 4, 14, 3, 5, 7, 9, 6, 6, 4, 8, 6, 7, 4, 4, 7, 3, 4, 7).$$

For a given degree sequence, an immediate and natural question is that whether there are other graphs with the same degree sequence. If there exists such a graph, the degree sequence may not be peculiar to it since there may exist a large number of graphs sharing the same degree sequence (Figure 2.3). For instance, there are 1.028.330.021.523.219.123.740.586.856.603.341.415.076.236[1] labeled simple graphs satisfying the degree sequence of the friendship network above. Hence, effective sampling of graphs from such a large set is not simple, and it is of great importance in many applications. The thesis aims to investigate different approaches proposed for generation of simple and bipartite graphs satisfying the given degree sequence, and proposes a uniform sampling and an exact counting algorithm for simple graphs.

1.1. Historical Background

The basis of random graphs dates back to the late 1950s with the study of famous mathematicians Erdös and Rényi [7]. They initiated a systematic study of random graphs with the papers published between the years 1959 and 1968, and introduced a random graph model named *Erdös-Rényi (ER)*. The notation $\mathcal{G}_{n,m}$ is used to represent the model with the parameters: n and m where n is the number of vertices and m is the number of edges. In the model, a graph is uniformly chosen at random from the set of all graphs having n vertices and m edges. In 1959, Gilbert [8] independently introduced another procedure, named *Gilbert model* $\mathcal{G}_{n,p}$ where connections between pair of vertices are determined with a probability p. Although a similar study was carried out earlier by Ray Solomonoff and Anatol Rapoport in 1951 [9], their work did not draw much attention at that time. By setting m to $\binom{n}{2}p$, it can be observed that the models $\mathcal{G}_{n,p}$ and $\mathcal{G}_{n,m}$ behaves similarly as n increases by the law of large numbers. The $\mathcal{G}_{n,p}$ model is more commonly used since dealing with the $\mathcal{G}_{n,m}$ is not much practical in mathematical analyses.

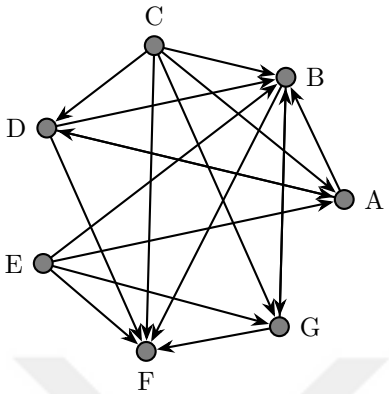
[1] In our knowledge, we are the first one to be able to compute this number exactly by using the algorithm that we have developed in Section (4.2.2). Our calculation closely matches the estimated result found by using the method proposed by Blitzstein and Diaconis [1].

Each vertex degree in the $\mathcal{G}_{n,p}$ model follows binomial distribution with parameters $n-1$ and p since a vertex can be connected to at most $n-1$ vertices, and it can be approximated by a Poisson distribution for large n values.

However, *Erdös-Rényi (ER)* model is not appropriate to capture various properties of real-world networks. For instance, in many social networks, individuals tend to cluster into smaller groups relative to the size of the whole network. Every individual in the society actually is a member of a certain community consisting of persons mostly knowing each other and sharing common areas of interests. The property of a graph, *clustering coefficient*, measures the extent to one's acquaintances are also acquaintances of each other, and many social networks exhibit high clustering coefficient contrary to the Erdös-Rényi (ER) random graphs. Moreover, any two persons who seem to be very distant from each other in the network can be actually connected with a short chain of people. Various studies like Milgram's experiment demonstrated that a person can reach another one in the network by means of a path involving around six people, and this phenomenon is generally referred to as *six degrees of separation* rule. Therefore, many real-world networks have a small average shortest path length. The networks combining these structural features are called as *small-world networks*. In 1998, Watts and Strogatz [10] proposed their random graph model, named *Watts-Strogatz (WS)* model [10], in order to construct and study small-world networks.

In the following years, it was noticed that Watts-Strogatz model was not sufficient to represent numerous real-world networks. Albert-László Barabási and his student Réka Albert showed that many complex networks such as World Wide Web, protein-protein interactions exhibit power-law distributions. That is, the probability of a vertex having a degree of k is proportional to $(1/k)^{\alpha}$ for some $\alpha > 1$, and those networks are referred to as *scale-free*. The first and most widely known procedure is *Barabási–Albert (BA)* model [11] for generation of scale-free networks, and many other models are later suggested for scale-free networks [12, 13]. Unlike to many other models, the scale-free networks contain *hubs*, which are high-degree vertices having much more connections than others in the network. It is one of the significant characteristics of scale-free networks because they can shorten the path length between any pair of vertices.

On the other hand, removing a few hubs may cause the network to be divided into disjoint components, and communications among vertices can be considerably affected.



		Managers						
		A	B	C	D	E	F	G
Managers	A	0	1	0	1	0	0	0
	B	0	0	0	0	0	1	1
	C	1	1	0	1	0	1	1
	D	1	1	0	0	0	1	0
	E	1	1	0	0	0	1	1
	F	0	0	0	0	0	0	0
	G	0	1	0	0	0	1	0

Figure 1.2. A subgraph of the Manager Network Dataset and Corresponding Binary Matrix

In recent years, the interest in random graphs has widely increased and several methods have been proposed for generation of graphs satisfying given degree sequences. These methods can be used as an alternative way for obtaining graph samples from a distribution. For instance, vertex degrees $(d_1, ..., d_n)$ of a graph are drawn from the desired distribution, and then uniformly a graph is selected from the set of graphs realizing the degree sequence $(d_1, ..., d_n)$ at random. In this way, samples distributed according to the desired distribution can be generated. Another motivation for studying random graphs with prescribed degree sequence might be to carry out hypotheses tests on a question arising in various fields. For instance, Krackhardt [14] collected data from the management team of a company by asking each manager whom he or she goes to request help for a problem or to give an advice. Managers' different attitudes towards their colleagues can be represented by a directed graph, and a part of this graph is depicted in Figure 1.2. A researcher may wonder whether there is a mutual interaction among managers in the company, and test the null hypothesis of tendency towards mutuality by uniformly generating samples satisfying the given degree sequence at random [15].

Furthermore, it may not be always possible to have enough knowledge about the whole network structure, some information related to vertex connections may be missing, and one can posses the degree sequence of the network. Consider an infectious disease transmitting from a person to person by direct or indirect contact in a society. Each individual can have roughly knowledge about the number of people he or she contacted in a certain time interval but may not express the real identities of every person. An epidemiologist may desire to construct a suitable contact graph to study the infectious scenarios of the disease on it [16].

Random graphs may be also used for enumeration of objects satisfying certain constrains. For example, a chemical molecule can be modeled by a graph structure by considering each atom as a vertex and bounds between atoms as edges. A molecule can have different structures denoted by the same molecular formula, which are called *isomers* of the molecule. If the valencies of atoms in the molecule are known, then isomers of the molecule can be counted by using random graphs [17].

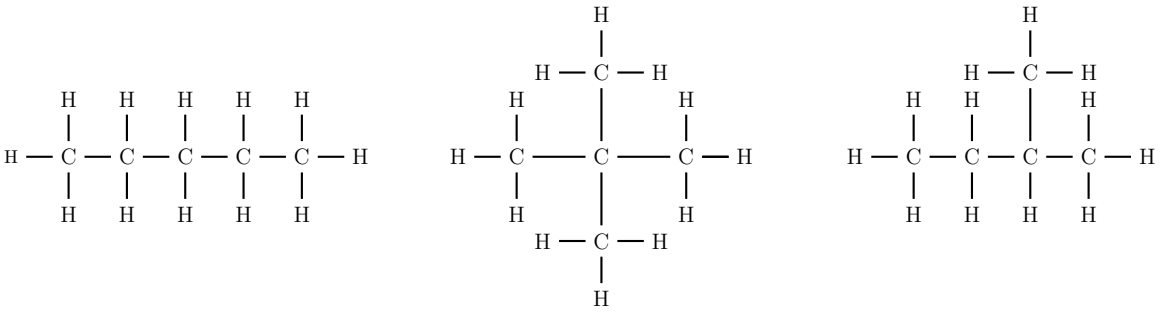


Figure 1.3. Three Structural Isomers of Pentane, C_5H_{12}

1.2. Organization of the Thesis and Contributions

The rest of the thesis is organized as follows: The fundamental definitions and notations for graph theory and probability theory are reviewed in Chapter 2. A natural and well-known edge-switching Markov chain method for simple graphs is described in Chapter 3.

Chapter 4 is devoted to direct construction algorithms, and two different approaches using Monte Carlo methods, importance sampling and sequential importance sampling, are presented in Section 4.1 for simple and bipartite graphs. Two well-known algorithms suggested by Havel-Hakimi and Erdös-Gallai to check whether a degree sequence is graphical or not is given in Sections 3.1 and 4.1, and the analogous theorem of Gale–Ryser for bipartite graphs is stated in Section 4.2. In Subsection 4.2.2, the main contribution of the thesis is described: a new uniform sampling and exact counting algorithm is proposed for simple graphs by transforming and adapting the algorithm given for bipartite graphs in Subsection 4.2.1. In Chapter 5, the applications of the presented algorithms are illustrated in several examples such as hypothesis testing, network analysis and graph enumeration. Finally, the thesis is concluded and the possible future works are stated in Chapter 6.

2. PRELIMINARIES

In this chapter, the fundamental definitions and notations for graph theory and probability theory which are needed for the subsequent sections are briefly reviewed. A reader who is unfamiliar with the concepts and who desires more detailed knowledge may check the references [18–23].

2.1. Basics of Graph Theory

A *graph* $G = (V, E)$ is an ordered pair of sets where V is the *vertex set* and E is the *edge set* such that $E \subseteq \{\{u, v\} : u, v \in V\}$. If the edge and vertex sets of a graph G are not explicitly written, they will be denoted by $V(G)$ and $E(G)$, respectively. Since only finite graphs are considered throughout the thesis, the vertex set V is finite. An edge is an unordered pair of vertices and denoted by $\{u, v\}$ or $u \sim v$. A graph is called *directed* or *digraph* if its edges have an orientation, and they are represented by the ordered pair (u, v) for some vertices $u, v \in V(G)$.

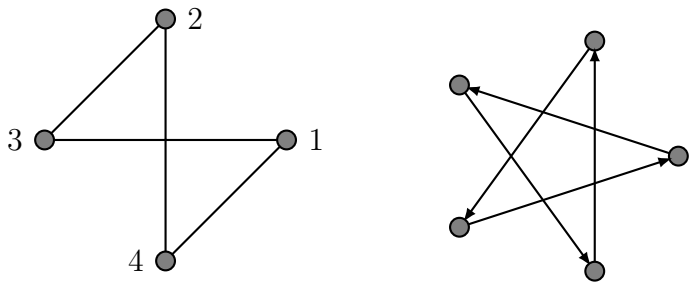


Figure 2.1. Undirected and Directed Graphs

For an edge $\{u, v\} \in E(G)$, it is said that u is *adjacent* to v or u is a neighbor of the vertex v. The set of all neighbors of the vertex v is denoted by $\mathcal{N}(v)$. A *loop* in a graph is an edge connecting a vertex to itself. *Multiple* edges are two or more edges connecting the same vertices. An *independent* set is a subset of the vertex set of a graph where any two vertices are not adjacent. For instance, the subsets $\{1, 3, 4, 6\}$ and $\{2, 5\}$ of the vertex set of the graphs in Figure 2.2 are independent sets.

Perfect matching of a collection of vertices is a set of edges where every vertex is adjacent to exactly one vertex. A partition P of a set S is a collection of non-empty subsets of S where $\bigcup_{\mathcal{X}\in P} \mathcal{X} = S$, and $\mathcal{X} \cap \mathcal{Y} = \emptyset$ for any $\mathcal{X}, \mathcal{Y} \in P$.

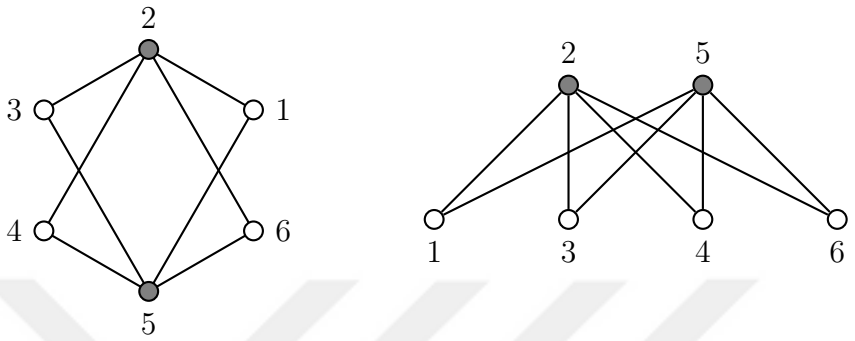


Figure 2.2. Isomorphic Bipartite Graphs

A graph with no edges and no vertices is called *null*, and a graph having at least one vertex with no edges is said to be *empty* graph. A *simple* graph is a graph containing no loops and no multiple edges. A graph is *bigraph* or *bipartite* if its vertices is partitioned into two disjoint independents sets V^+ and V^-. A graph is called *labeled* if its vertices are assigned to an element from a non-empty set. The degree of a vertex v of an undirected graph is the number of edges adjacent to v, and denoted by $deg(v)$. The degree sequence $\boldsymbol{d}$ of a labeled undirected graph G with vertex set $V(G) = \{1, ..., n\}$ is a sequence $(d_1, ..., d_n)$ in which the degree of a vertex $v \in V(G)$ is equal to d_v. In Figure 2.3, two different (i.e. non-isomorphic) simple graphs having the same degree sequence $(3, 2, 2, 3, 2, 2)$ are depicted.

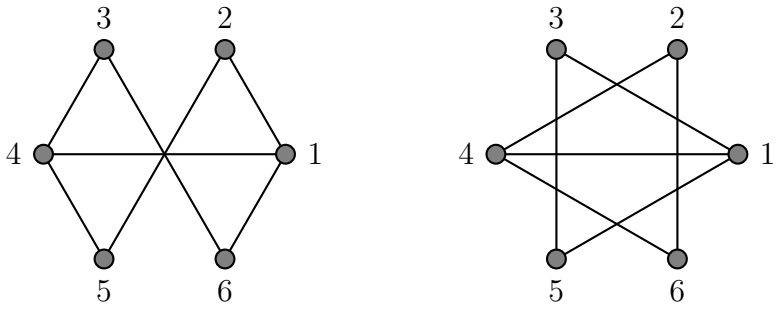


Figure 2.3. Two Non-isomorphic Graphs Sharing the Same Degree Sequence

Two simple graphs G and $\widetilde{G}$ are called *isomorphic*, and denoted by $G \simeq \widetilde{G}$ if there exists an isomorphism $f : V(G) \to V(\widetilde{G})$ such that $\{u, v\} \in E(G)$ if and only if $\{f(u), f(v)\} \in E(\widetilde{G})$. If the graphs G and $\widetilde{G}$ are labeled with labeling functions l and $\widetilde{l}$, the additional condition that $l(v) = \widetilde{l}(f(v))$ for all $v \in V(G)$ must be satisfied.

Let $\boldsymbol{d} = (d_1, ..., d_n) \in \mathbb{N}^n$ be a non-increasing degree sequence, and the *conjugate* of $\boldsymbol{d}$ is denoted by $\boldsymbol{d}^*$, and defined by $(d_1^*, ..., d_n^*)$ where $d_i^* = |\{d_k : d_k \geq i\}|$. For any sequence $\boldsymbol{d} = (d_1, ..., d_n) \in \mathbb{N}^n$, the functions $\oplus_{i_1,...,i_k}$ and $\ominus_{i_1,...,i_k}$ are defined by

$$\oplus_{i_1,...,i_k}\boldsymbol{d} = \begin{cases} d_i + 1 & for\ i \in \{i_1, ..., i_k\} \\ d_i & otherwise \end{cases} \qquad \ominus_{i_1,...,i_k}\boldsymbol{d} = \begin{cases} d_i - 1 & for\ i \in \{i_1, ..., i_k\} \\ d_i & otherwise \end{cases}$$

for distinct indices $i_1, ..., i_n \in \{1, ..., n\}$

2.2. Basics of Probability Theory

Let X be a set, and its power set is denoted by $\mathcal{P}(\mathsf{X})$, which contains all subsets of X. A subset $\mathcal{X} \in P(\mathsf{X})$ is said to be a *σ-algebra*, if $\mathsf{X} \in \mathcal{X}$, any set $A \in \mathcal{X}$ implies $\mathsf{X} \backslash A \in \mathcal{X}$, and the union $\bigcup_{i=1} A_i \in \mathcal{X}$ for $A_1, A_2, \ldots \in \mathcal{X}$. If X is a topological space, then the σ-algebra generated by the open sets of X is called the *Borel* σ-algebra on X, and denoted by $\mathcal{B}(\mathsf{X})$.

A *measurable* space $(\mathsf{X}, \mathcal{X})$ is an ordered pair where $\mathcal{X}$ is a σ-algebra of X, and the elements of $\mathcal{X}$ is called *measurable sets*. A *measure* μ on $(\mathsf{X}, \mathcal{X})$ is a function from $\mathcal{X}$ to $\bar{\mathbb{R}}_+$ satisfying $\mu(\emptyset) = 0$, $\mu(A) \geq 0$ for all $A \in \mathcal{X}$ and $\mu(\cup_i A_i) = \sum_i \mu(A_i)$ for all countable collections of pairwise disjoint sets $\{A_i\}_{i>0}$ where $A_i \in \mathcal{X}$. A measure $\mathbb{P}$ is a *probability measure*, if $\mathbb{P}(\mathsf{X}) = 1$. An ordered pair $(\mathsf{X}, \mathcal{X}, \mathbb{P})$ is said to be a *probability space* where a probability measure $\mathbb{P}$ is defined on the measurable space $(\mathsf{X}, \mathcal{X})$.

Let $(\mathsf{X}, \mathcal{X})$, and $(\mathsf{Y}, \mathcal{Y})$ be measurable spaces. A function $X : \mathsf{X} \longrightarrow \mathsf{Y}$ is called *measurable* if the preimage $X^{-1}(B) = \{a : X(a) \in B\}$ is in $\mathcal{X}$ for every $B \in \mathcal{Y}$.

If $\mathcal{Y}$ is equal to $\mathcal{B}(\bar{\mathbb{R}})$, then the function X is called $\mathcal{X}$-measurable where the set $\mathcal{B}(\bar{\mathbb{R}})$ denotes the Borel σ-algebra on $\bar{\mathbb{R}}$, and $\mathcal{X}$-measurable functions are named as *Borel functions*. A function X from a probability space $(\mathsf{X}, \mathcal{X}, \mathbb{P})$ to a measurable space $(\mathsf{Y}, \mathcal{Y})$ is called *random variable* if X is measurable.

For a random variable $X : \mathsf{X} \longrightarrow \mathsf{Y}$, the measure μ on $(\mathsf{Y}, \mathcal{Y})$ defined by

$$\mu(B) := \mathbb{P}\left(X^{-1}(B)\right) = \mathbb{P}\{X \in B\}, \quad B \in \mathcal{Y}$$

is called the *distribution* of the random variable X. Let μ and ν be measures on a measurable space $(\mathsf{X}, \mathcal{X})$, then μ is said to be *absolutely continuous* with respect to ν if $\nu(A) = 0$ implies $\mu(A) = 0$ for every set $A \in \mathcal{X}$.

Let $(\mathsf{X}, \mathcal{X})$, and $(\mathsf{Y}, \mathcal{Y})$ be measurable spaces. A function $K : \mathsf{X} \times \mathcal{Y} \longrightarrow \bar{\mathbb{R}}_+$ is called *transition kernel* from $(\mathsf{X}, \mathcal{X})$ to $(\mathsf{Y}, \mathcal{Y})$ if the following conditions are satisfied:

(i) the mapping $x \longmapsto K(x, B)$ is $\mathcal{X}$-measurable for every element $B \in \mathcal{Y}$
(ii) the mapping $B \longmapsto K(x, B)$ is a measure on $(\mathsf{Y}, \mathcal{Y})$ for every element $x \in \mathsf{X}$

A transition kernel K from $(\mathsf{X}, \mathcal{X})$ to $(\mathsf{X}, \mathcal{X})$ satisfying $K(x, \mathsf{X}) = 1$ for every $x \in \mathsf{X}$ is said to be a *Markov kernel* on $(\mathsf{X}, \mathcal{X})$. Let μ be a measure on a measurable space $(\mathsf{X}, \mathcal{X})$, and the measure μK on $(\mathsf{Y}, \mathcal{Y})$ is given by

$$\mu K(B) := \int_{\mathsf{X}} \mu(da) K(a, B) \quad for \quad B \in \mathcal{Y}$$

and the measure $\mu \otimes K$ on the product space $(\mathsf{X} \times \mathsf{Y}, \mathcal{X} \otimes \mathcal{Y})$ is defined by

$$(\mu \otimes K)(C) := \int \int_C \mu(da) K(a, db) \quad for \quad C \in \mathcal{X} \otimes \mathcal{Y} \tag{2.1}$$

Let K_1 and K_2 be transition kernels from $(\mathsf{X}, \mathcal{X})$ to $(\mathsf{Y}, \mathcal{Y})$, and from $(\mathsf{Y}, \mathcal{Y})$ to $(\mathsf{Z}, \mathcal{Z})$, respectively. The product of kernels, K_1K_2, defined by

$$K_1K_2(a, C) := \int_{\mathsf{Y}} K_1(a, db)K_2(b, C) \quad a \in \mathsf{X}, C \in \mathcal{Z}$$

is a transition kernel from $(\mathsf{X}, \mathcal{X})$ to $(\mathsf{Z}, \mathcal{Z})$. The transition kernel $K_3 \otimes K_4$ from $(\mathsf{X}, \mathcal{X})$ to $(\mathsf{Y} \times \mathsf{Z}, \mathcal{Y} \times \mathcal{Z})$ is given by

$$(K_3 \otimes K_4)(a, D) := \int \int_D K_3(a, db)K_4(a, b, dc), \quad a \in \mathsf{X}, \ D \in \mathcal{Y} \times \mathcal{Z}$$

where K_3 and K_4 are kernels from $(\mathsf{X}, \mathcal{X})$ to $(\mathsf{Y}, \mathcal{Y})$, and from $(\mathsf{X} \times \mathsf{Y}, \mathcal{X} \times \mathcal{Y})$ to $(\mathsf{Z}, \mathcal{Z})$, respectively. For a transition kernel K on $(\mathsf{X}, \mathcal{X})$, its powers $K^0 = I$, $K^1 = K$, $K^2 = KK$, $K^3 = KK^2$,... are the transition kernels on $(\mathsf{X}, \mathcal{X})$ where the identity kernel $I(x, A) = \delta_x(A)$ is defined by

$$\delta_x(A) = \begin{cases} 1, & if \ x \in A \\ 0, & if \ x \notin A \end{cases}$$

where δ_x is generally referred to as *Dirac measure*.

Let T be an arbitrary countable or uncountable index set, and the collection of X-valued random variables $\{X_i : i \in \mathcal{T}\}$ is called a X-valued *stochastic process*. A family of σ-algebras $\{F_i : i \in \mathcal{T}\}$ on X is called *filtration of* $(\mathsf{X}, \mathcal{X})$ if $\mathcal{F}_i \subseteq \mathcal{X}$ for all $i \in \mathcal{T}$, and $\mathcal{F}_i \subseteq \mathcal{F}_j$ whenever $i < j$ for $i, j \in \mathcal{T}$. A filtered probability space is a quadruple $(\mathsf{X}, \mathcal{X}, \mathcal{F}, \mathbb{P})$ where $\mathcal{F} = \{\mathcal{F}_i : i \in \mathcal{T}\}$ is a filtration of $(\mathsf{X}, \mathcal{X})$. A stochastic process $\{X_i : i \in \mathcal{T}\}$ is $\mathbb{F}$-*adapted*, if each random variable X_i is $\mathcal{F}_i$-measurable for all $i \in \mathcal{T}$.

3. MARKOV CHAIN MONTE CARLO METHODS

There are various types of methods proposed for sampling graphs with a given degree sequence, and most of them can be roughly grouped into two categories: direct graph construction algorithms from scratch by adding edges according to a certain rule, or methods relying on the conversion of an existing graph into a new one by successively removing and adding edges without changing the degrees of vertices. This chapter considers the latter case, and investigates an approach based on a stochastic process in order to generate uniformly distributed simple graphs satisfying a particular degree sequence.

3.1. Switch Markov Chain for Simple Graphs

A sequence $\boldsymbol{d} = (d_1, ..., d_n)$ is called *graphical* if there exists a labeled graph G with the vertex set $\{1, ..., n\}$ where vertex v has a degree d_v, and the graph G is called *realization* of the degree sequence $\boldsymbol{d}$. Let $\mathcal{G}^{\boldsymbol{d}}$ be the set containing all realizations of the graphical degree sequence $\boldsymbol{d}$.

The *switch chain* model is one of the mostly used Markov chain Monte Carlo methods to generate samples realizing a given graphical degree sequence. The switch Markov chain $\mathcal{M}_{\boldsymbol{d}}$ is defined on the set $\mathcal{G}^{\boldsymbol{d}}$ as follows: Let G be a realization of the specified degree sequence, then do nothing with a probability $\frac{1}{2}$. Otherwise, uniformly select two edges $\{v_1, v_2\}$, $\{v_3, v_4\}$ having no common vertex, and switch the edges with a uniformly chosen perfect matching $\mathcal{M}$ of the set $\{v_1, v_2, v_3, v_4\}$ if $\mathcal{M} \cap E(G) = \emptyset$ so that vertex degrees of the graph remain the same, and simple graph assumption is not violated. If $\mathcal{M} \cap E(G) \neq \emptyset$, stay at the same realization G. In the chain, each move from the previous graph G to the current realization is called a *switching* step, and the more detailed description of the algorithm is given in Figure 3.3.

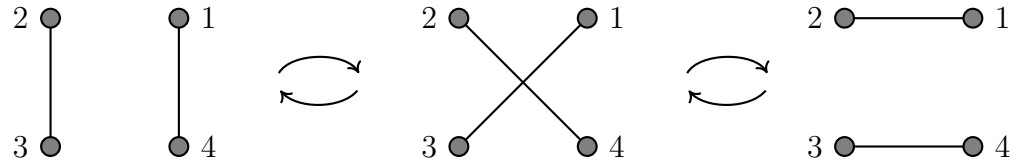


Figure 3.1. Illustration of Edge Switching Steps

Note that an initial graph realizing a given degree sequence $\boldsymbol{d}$ is needed in order to carry out the switching steps, and it can be constructed with the well-known algorithm of Havel and Hakimi. Beginning with an empty graph, a vertex v is chosen randomly, and it is attached to $deg(v)$ distinct vertices $u_1, ..., u_{deg(v)}$ having the largest $deg(v)$ degrees in the sequence $\boldsymbol{d} = (d_1, ..., d_n)$. Then, the degree sequence $\boldsymbol{d}$ is updated to $\ominus_{v,u_1,..,u_{deg(v)}}\boldsymbol{d}$, and the process is repeated until all elements of the degree sequence $\boldsymbol{d}$ becomes zero.

Theorem 3.1. *(Havel-Hakimi) Let $\boldsymbol{d} = (d_1, ..., d_n) \in \mathbb{N}^n$ be a degree sequence where $d_i < n$ for all $1 \leq i \leq n$, and $\widetilde{\boldsymbol{d}}$ be a degree sequence obtained from $\boldsymbol{d}$ by subtracting 1 from d_j highest elements (other than j) in $\boldsymbol{d}$, and by letting $\widetilde{d}_j = 0$ for some $j \in \{1, ..., n\}$. Then $\boldsymbol{d}$ is graphical if and only if $\widetilde{\boldsymbol{d}}$ is graphical. Moreover, if $\boldsymbol{d}$ is graphical, there is a realization of $\boldsymbol{d}$ containing edges between the vertex j and d_j highest degree vertices (other than vertex j).*

Proof. Assume that the sequence $\widetilde{\boldsymbol{d}}$ is graphical, then it has a realization $\widetilde{G}$ in which the degree of the vertex v is equal to 0 for some $v \in \{1, ..., n\}$. Let $\mathcal{S}$ be a set containing vertices (except v) having the d_v highest degrees in $\boldsymbol{d}$. Then, a realization G of $\boldsymbol{d}$ can be obtained by simply adding the edges $\{v, s\}$ for all $s \in \mathcal{S}$ to the realization $\widetilde{G}$.

Suppose $\boldsymbol{d}$ is a graphical degree sequence, and v is a vertex where $deg(v) > 0$ in a realization G of $\boldsymbol{d}$. If G contains the edges $\{v, s\}$ for all $s \in \mathcal{S}$, then a realization of $\widetilde{\boldsymbol{d}}$ can be obtained by removing the edges $\{v, s\}$ for all $s \in \mathcal{S}$. If $\{v, w\} \in E(G)$ for some vertex $w \notin \mathcal{S}$, then there exist a vertex $\tilde{s} \in \mathcal{S}$ such that $\{v, \tilde{s}\} \notin E(G)$. If the degree of the vertex w is equal to the degree of $\tilde{s}$ then switch the vertices w and $\tilde{s}$.

Thus, a realization containing all of the edges $\{v, s\}$ for every $s \in \mathcal{S}$ is obtained, and apply the previous case. If the degree of w is less than the degree of $\tilde{s}$, then there exists a vertex $u \in \mathcal{N}(\tilde{s}) \backslash (\mathcal{N}(w) \cup \{w\})$. Replace the edges $\{v, w\}$, $\{\tilde{s}, u\}$ with $\{v, \tilde{s}\}$ and $\{w, u\}$, so a realization of $\boldsymbol{d}$ containing all of the edges $\{v, s\}$ for every $s \in \mathcal{S}$ is attained, and then apply the first case. Note that the degree of w cannot be greater than the degree of $\tilde{s}$ because $\tilde{s} \in \mathcal{S}$, and $w \notin \mathcal{S}$ by assumption. The same arguments (if necessary) can be repeated until a realization of $\boldsymbol{d}$ containing all of the edges $\{v, s\}$ for $s \in \mathcal{S}$ is obtained. □

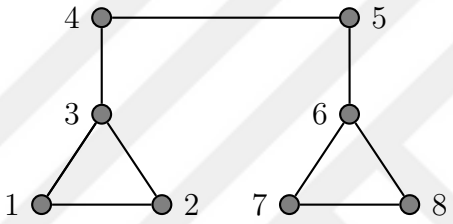


Figure 3.2. A Realization of the Sequence $(2, 2, 3, 2, 2, 3, 2, 2)$

Havel-Hakimi algorithm can be also considered as an alternative to the Erdös-Gallai Theorem 4.1 in testing whether a given degree sequence is graphical. For instance, one can obtain the following sequence of graphical degree sequences for $(3, 4, 5, 2, 2, 2)$.

$$(3, 4, 5, 2, 2, 2) \to (3, 3, 4, 0, 2, 2) \to (2, 3, 3, 0, 2, 0) \to (1, 0, 2, 0, 1, 0) \to (1, 0, 1, 0, 0, 0)$$

with the chosen indices $4, 6, 2$, and 5, respectively. Since the sequence $(1, 0, 1, 0, 0, 0)$ is graphical, $(3, 4, 5, 2, 2, 2)$, and the intermediate sequences are also graphical.

Notice that any realization of a given graphical degree sequence cannot be generated with this method. For instance, consider the realization of the degree sequence $(2, 2, 3, 2, 2, 3, 2, 2)$ in Figure 3.2. In the first step, if a vertex of degree 2 is chosen, then the vertex must be adjacent to both of vertices having degree 3, but there is only one edge in the graph between any pair vertices of degree 2 and 3.

Initially, a vertex having degree 3 cannot be also taken because there is no edge between the vertices of degree 3. Thus, the graph drawn in Figure 3.2 cannot be constructed by Havel-Hakimi algorithm.

Input: A graph G^0 satisfying the degree sequence $\boldsymbol{d} = (d_1, ..., d_n)$
Output: A sample G^K realizing the degree sequence $\boldsymbol{d} = (d_1, ..., d_n)$

for $k = 1$ to K **do**
  $r \leftarrow$ a uniformly chosen number from (0,1)
  if $r > \frac{1}{2}$ **then**
    Uniformly choose two random non-adjacent edges $\{v, u\}$ and $\{x, y\}$.
    Choose a perfect matching M of $\{v, u, x, y\}$
    if $M \cap E(G) = \emptyset$ **then**
      Construct G^{k+1} by removing the edges $\{v, u\}$, $\{x, y\}$ and adding
      the edges of M
    end if
  end if
end for

Figure 3.3. The Switch Markov chain $\mathcal{M}_{\boldsymbol{d}}$ on the set $\mathcal{G}^{\boldsymbol{d}}$

Let $(\mathsf{Y}, \mathcal{Y}, \mathcal{F}, \mathbb{P})$ be a filtered probability space, and K be a Markov transition kernel on a measurable space $(\mathsf{X}, \mathcal{X})$. An $\mathcal{F}$-adapted X-valued stochastic process $\{X_t\}_{t\in\mathcal{T}}$ is called *Markov chain* with respect to the filtration $\mathcal{F}$, and the transition kernel K if the process $\{X_t\}_{t\in\mathcal{T}}$ satisfies

$$\mathbb{P}(X_{t+1} \in B|\mathcal{F}_t) = K(X_t, B) \tag{3.1}$$

for all $t \in T$, and sets $B \in \mathcal{X}$.

If the measurable space $(\mathsf{X}, \mathcal{P}(\mathsf{X}))$ is countable, then the Markov property (3.1) can be rewritten as

$$\mathbb{P}(X_{t+1} = x^{(t+1)} | X_t = x^{(t)}, ..., X_0 = x^{(0)}) = \mathbb{P}(X_{t+1} = x^{(t+1)} | X_t = x^{(t)}) \tag{3.2}$$

where $x^{(i)} \in \mathsf{X}$ for all $i \in \mathcal{T}$ and the transition kernel can be represented by a matrix $K = (k_{ij})$ where $k_{ij} = \mathbb{P}(X_{t+1} = x_j | X_t = x_i)$.

Note that the switch chain $\mathcal{M}_{\boldsymbol{d}}$ described above is defined on the finite state space consisting of the simple graphs realizing the graphical degree sequence $\boldsymbol{d} = (d_1, \ldots, d_n)$. Given the current graph state, the future graphs in the chain are independent of the past states, so the chain $\mathcal{M}_{\boldsymbol{d}}$ satisfies the Markov property (3.2). Since the transition kernel K remains the same for each index $t \in \mathcal{T}$, the process is also a *time-homogeneous* Markov chain.

A probability measure π on a countable state space X is a *stationary distribution* of the Markov chain $\{X_t\}_{t \in \mathcal{T}}$ if it satisfies $\pi(y) = \sum_{x \in \mathsf{X}} \pi(x) K(x, y)$ for all states $x, y \in \mathsf{X}$. For an initial probability distribution μ, if the measure μK^n on $(\mathsf{X}, \mathcal{X})$ converges to a *limiting distribution* π as n increases, then π must be the stationary distribution of the chain, and if the limiting distribution π is unique, it is independent of the initial measure μ [24]. By using this fact, one can generate approximate samples from a desired distribution by constructing a Markov chain converging to this distribution. Then, initially chosen samples from any arbitrary measure are (almost) distributed according to the limiting distribution after taking sufficient number of steps in the chain. However, a constructed Markov chain may not be convergent or it may have more than one stationary distribution. The following theorem gives the sufficient conditions for a Markov chain to guarantee that it has a unique stationary distribution and it converges to the stationary distribution.

Theorem 3.2. *[24] If a Markov chain $\{X_t\}_{t\in\mathcal{T}}$ is irreducible, positive recurrent and aperiodic on a countable state space X with transition kernel K, then it has a unique stationary distribution π, and*

$$\sup_{x\in\mathsf{X}} \left|\mu K^n(x) - \nu K^n(x)\right| \longrightarrow 0, \quad as\ n \longrightarrow \infty$$

for any initial probability measures μ and ν. Let μ be δ_x and ν be π, then it can be obtained that

$$\sup_{x\in\mathsf{X}} \left|K^n(y,x) - \pi(x)\right| \longrightarrow 0, \quad as\ n \longrightarrow \infty \quad for\ any\ y \in \mathsf{X}$$

A state $x \in \mathsf{X}$ is *visitable* from another one $y \in \mathsf{X}$ in a countable state space, if there exists $k \in \mathbb{N}$ such that $K^k(y,x) > 0$, and a Markov chain is called *irreducible* if every state is visitable from any other state. A state x is said to be *recurrent* if it is always visited in a finite time with probability 1 by starting at the same state x, and it is *positive recurrent* if the expected number of steps required to return the same state is finite. Another characteristic of a state x is its *periodicity* which is defined by the greatest common divisor of the set $\{n > 0 : K^n(x,x)\} > 0$, and it is said to be *aperiodic* if the period of the state is one. A Markov chain is called positive recurrent, and aperiodic if all states of the process are positive recurrent, and aperiodic, respectively.

The following corollary shows that the switch chain $\mathcal{M}_{\boldsymbol{d}}$ is irreducible, positive recurrent and aperiodic, so it converges to a unique stationary distribution, and it is uniform over the set of all realizations of the degree sequence $\boldsymbol{d}$. Hence, the switch chain $\mathcal{M}_{\boldsymbol{d}}$ can be used to generate samples (approximately) uniformly distributed over the set $\mathcal{G}^{\boldsymbol{d}}$.

Corollary 3.3. *Let $\boldsymbol{d} \in \mathbb{N}^n$ be a graphical degree sequence. The switch Markov chain $\mathcal{M}_{\boldsymbol{d}}$ converges to the uniform stationary distribution over $\mathcal{G}^{\boldsymbol{d}}$ regardless of any initial realization of the degree sequence $\boldsymbol{d} = (d_1, ..., d_n)$.*

Proof. It can be seen that one can reach any realization of the sequence $\boldsymbol{d}$ from another one with a sequence of edge switching operations [25] so the chain is irreducible. Since the state space $\mathcal{G}^{\boldsymbol{d}}$ is finite, the irreducible Markov chain $\mathcal{M}_{\boldsymbol{d}}$ is also positive recurrent [24]. The chain can stay at the same state with a probability greater than $1/2$ by the third and sixth steps of the algorithm in Figure 3.3, so it is aperiodic. By Theorem 3.2, the chain has a unique stationary distribution π satisfying $K^T\pi = \pi$ where K is the state transition matrix of the chain. K is symmetric due to the construction of the chain so $K^T = K$, and 1 is an eigenvalue of the matrix K. Hence, π is the uniform distribution over $\mathcal{G}^{\boldsymbol{d}}$ since the each row sum of the matrix K equals to 1. □

The *mixing time* of a Markov chain is the number of steps required to be sure that the distribution of the states of the chain gets closer enough to the stationary distribution. A useful Markov chain should posses the *rapid mixing* property so that a sample from the stationary distribution can be generated in a polynomial time with respect to the input size. In 1999, Kannan, Tetali and Vempala [26] proposed an edge switching Markov chain method, and they conjectured that the chain *mixes rapidly* for any degree sequence, but they could give a proof only for regular bipartite graphs. Later, Cooper, Dyer and Greenhill [27] proved it for regular undirected graphs, and in 2011 Greenhill [28] showed that the switch chain is rapidly mixing for regular directed graphs. In 2017, Greenhill and Sfragara [29] proved that the switch chain for undirected graphs with a degree sequence $(d_1, ..., d_n)$ is rapidly mixing if $d_i \geq 1$ for all $1 \leq i \leq n$, and the maximum degree d_{max} satisfies $3 \leq d_{max} \leq \frac{1}{3}\sqrt{\sum_{i=1}^{n} d_i}$. Many other proofs have been also provided for special types of graphs and degree sequences [30–33], but it is still an open question for the general case.

4. DIRECT CONSTRUCTION METHODS

Bipartite graphs are another important mathematical structures which are used to model interactions between different two groups of objects, and naturally arise in many applications such as affiliations networks. This chapter is devoted to direct construction methods for simple and bipartite graphs with given degree sequences.

4.1. Weighted Sampling Methods

Approximately uniform samples can be generated by using the edge-switching based Markov chain method introduced in the previous section. However, the unknown mixing time of the chain is its main disadvantage, and directly obtaining uniformly distributed samples over the set of realizations of a given degree sequence is not simple due to the complicated interactions between the vertices of a graph. This section considers different Monte Carlo approaches allowing one to estimate the properties of the uniform distribution over the set of all graphs satisfying a given degree sequence.

4.1.1. Importance Sampling

The expected value of random variables are encountered in many fields, and most problems can be also restated as an expectation of certain functions. However, in most cases it is not always simple to analytically evaluate these expectations due to the various reasons such as high dimensionality of the problem space or the nature of functions. In such cases, Monte Carlo methods become very useful in the computation of expectations when the other approaches are inadequate.

Let $(\mathsf{X}, \mathcal{X})$ be a measurable space, Ξ be a random variable taking values in $(\mathsf{X}, \mathcal{X})$, and μ be the probability distribution of Ξ.

Then, the expected value of a bounded $\mathcal{X}$-measurable function f with respect to μ is defined by

$$\mathbb{E}_\mu[f(\Xi)] := \int_{\mathsf{X}} f(\xi)\mu(d\xi) \tag{4.1}$$

and the Monte Carlo estimator for the expectation of $\mathbb{E}[f(\Xi)]$ is given by

$$\widehat{f}_{\mu,N} = \frac{1}{N}\sum_{i=1}^{N} f(\xi_i) \tag{4.2}$$

where $\xi_1, ..., \xi_N$ are independent and identically distributed (iid) random variables from the distribution μ. The strong law of large numbers implies that the estimator $\widehat{f}_{\mu,N}$ converges almost surely to the expectation $\mathbb{E}_\mu[f(\Xi)]$ as N increases.

Importance sampling is a classical Monte Carlo method in which the expected value of a function with respect to a probability measure μ is approximated by using samples drawn from another probability distribution. In many situations, obtaining samples from the desired distribution μ turns into a challenging problem. The main idea of *importance sampling* relies on the change of probability measures so that samples can be drawn from a more appropriate measure ν rather than the distribution μ. Let the *target distribution* μ be absolutely continuous with respect to the *proposal distribution* ν and the expectation in (4.1) can be rewritten as

$$\mathbb{E}_\mu(f(\Xi)) := \int_{\mathsf{X}} f(\xi)\mu(d\xi) = \int_{\mathsf{X}} f(\xi)\frac{d\mu}{d\nu}(\xi)\nu(d\xi) = \int_{\mathsf{X}} f(\xi)w(\xi)\nu(d\xi) = \mathbb{E}_\nu[f(\Xi)w(\Xi)]$$

where f is bounded $\mathcal{X}$-measurable function, and the weight $w(\xi)$ is defined by $\frac{d\mu}{d\nu}(\xi)$ which is the Radon-Nikodym derivative of μ with respect to the measure ν. The estimator is given by

$$\widehat{f}_{\nu,N} = \frac{1}{N}\sum_{i=1}^{N} f(\xi^{(i)})\frac{d\mu}{d\nu}(\xi^{(i)})$$

where $\xi^{(i)} \sim \nu$ for $1 \leq i \leq N$.

Recall that $\mathcal{G}^{\boldsymbol{d}}$ is the set of simple graphs realizing the degree sequence $\boldsymbol{d} = (d_1, ..., d_n)$, and let μ be the uniform distribution over the measurable space $(\mathcal{G}^{\boldsymbol{d}}, \mathcal{P}(\mathcal{G}^{\boldsymbol{d}}))$. Since directly obtaining samples from the distribution μ is not simple, the importance sampling idea can be applied for this problem. Blitzstein and Diaconis [1] suggested a proposal distribution σ and designed an effective algorithm to generate samples from the measure σ. The more detailed description of the method is given in Figure 4.1 below.

Input: A graphical degree sequence $\boldsymbol{d} = (d_1, ..., d_n)$, and empty edge set $\mathcal{E}$

Output: An ordered edge set $\mathcal{E}$

1: If $\boldsymbol{d} = (0, ..., 0)$, return $\mathcal{E}$

2: Choose the least index v where d_v is the smallest non-zero element

3: Compute the candidate set $\mathcal{C}_v = \{w \neq v\text{: } \{v, w\} \notin \mathcal{E} \text{ and } \ominus_{v,w}\boldsymbol{d} \text{ is graphical}\}$

4: Choose an index $u \in \mathcal{C}_v$ with probability proportional to d_u

5: Add the edge $\{v, u\}$ to $\mathcal{E}$, and update $\boldsymbol{d}$ to $\ominus_{v,u}\boldsymbol{d}$.

6: Repeat the steps $3 - 5$ until the degree of v equals to 0

Figure 4.1. Sampling Algorithm for Simple Graphs Realizing the Degree Sequence

Let m be the index of the smallest non-zero element in the graphical degree sequence $\boldsymbol{d} = (d_1, ..., d_n)$, and let $\boldsymbol{d} = \boldsymbol{d}^{(0)}, \boldsymbol{d}^{(1)}, ..., \boldsymbol{d}^{(k)}$ be the graphical degree sequences where each sequence $\boldsymbol{d}^{(i)}$ is obtained by subtracting 1 from the sequence $\boldsymbol{d}^{(i-1)}$ at coordinates v_i and m for distinct indices $v_1, ..., v_k, m \in \{1, ..., n\}$ and $k \leq d_m^{(0)}$. Then, there exists a realization of $\boldsymbol{d}^{(0)}$ containing all of the edges $\{v_1, m\}, ..., \{v_k, m\}$, and a realization of $\boldsymbol{d}^{(k)}$ containing none of these edges [1]. Hence, the candidate set in the third step always contains at least one possible index, and the algorithm in Figure 4.1 runs until the sequence $\boldsymbol{d}$ equal to $(0, ..., 0)$ without getting stuck. The algorithm can also produce any realization of a given degree sequence with a positive probability.

In the third step, the graphicality of a degree sequence can be tested by checking the necessary and sufficient conditions given in the well-known Erdös-Gallai theorem.

Theorem 4.1. *(Erdös-Gallai). Let* $\boldsymbol{d} = (d_1, ..., d_n) \in \mathbb{N}^n$ *be a degree sequence with* $d_1 \geq d_2 \geq ... \geq d_n$*. Then the sequence* $\boldsymbol{d} = (d_1, ..., d_n)$ *is graphical if and only if* $\sum_i^n d_i$ *is even and*

$$\sum_{i=1}^{k} d_i \leq k(k-1) + \sum_{i=k+1}^{n} min(k, d_i) \ for \ each \ k \in \{1, ..., n\} \tag{4.3}$$

Proof. [34] For the necessity part, suppose the sequence $\boldsymbol{d}$ is graphical. The degree sum $\sum_i^n d_i$ is even since the sum of degrees is equal to twice the number of edges. Let $\mathcal{K}$ be a set of k vertices in a realization of $\boldsymbol{d}$. There are at most $\binom{k}{2}$ edges among vertices in $\mathcal{K}$, and any vertex $v \notin \mathcal{K}$ can have at most $\min(k, deg(v))$ edges from v into the set $\mathcal{K}$. Hence, the inequalities in (4.3) are satisfied for each $k \in \{1, ..., n\}$.

Let a *subrealization* of a given non-increasing degree sequence $\boldsymbol{d} = (d_1, ..., d_n)$ be a graph with vertex set $\{1, ..., n\}$ such that $deg(v) \leq d_v$ for all $v \in \{1, ..., n\}$. The *critical index* r of a subrealization is the maximum index value satisfying the equality $deg(v) = d_v$ for all $1 \leq v < r$. The proof of the sufficiency side of the theorem is based on the successive construction of subrealizations so that the difference between $deg(v)$ and d_v is decreased in each subrealization, and finally a graph satisfying the given degree sequence $\boldsymbol{d}$ will be obtained.

Initially, there is a subrealization containing no edge, so the critical index value r is less than n if the entries of the degree sequence $\boldsymbol{d}$ is not all zero, otherwise the proof completes. Let $\mathcal{S}$ be a set of vertices defined by $\{r+1, ..., n\}$. Note that $\mathcal{S}$ is an independent set since the initial subrealization does not contain any edge between the vertices in $\mathcal{S}$. Throughout the proof, the set $\mathcal{S}$ will remain independent in each subrealization. Let G be a subrealization of the degree sequence $\boldsymbol{d}$, and $\boldsymbol{r}$ be a critical index of it. Then,

Case 1, if there exists some index $v > r$ such that $\{v, r\} \notin \mathcal{E}$ and $deg(v) < d_v$, then add the edge $\{r, v\}$ into $E(G)$. Hence, $deg(r)$ is increased, and the set $\mathcal{S}$ remains independent.

Case 2, if there exists $v < r$ such that $\{v, r\} \notin \mathcal{E}$, then there exists some vertex $u \in V(G)$ such that $\{v, u\} \in E(G)$ and $\{u, r\} \notin E(G)$ since $deg(v) = d_v \geq d_r > deg(r)$. If $d_r - deg(r) \geq 2$, then remove the edge $\{v, u\}$, and add the edges $\{v, r\}$, $\{r, u\}$, so that $deg(r)$ is increased by 2. If $d_r - deg(r) = 1$, then there exists some vertex $w > r$ such that $d_w > deg(w)$ since $\sum_{i=1}^{n} d_i - \sum_{i=1}^{n} deg(i)$ is even and $d_i = deg(i)$ for all $i < r$. If there is not any edge between the vertices r and w, then *Case* 1 can be applied. Otherwise, replace the edges $\{r, w\}$, $\{v, u\}$ with $\{v, r\}$ and $\{u, r\}$.

Case 3, if $\{1, 2, ..., r-1\} \in \mathcal{N}(r)$, and there exists some $u > r$ such that $d_u \neq \min\{d_u, r\}$. In a subrealization, $deg(u) \leq d_u$ by assumption, and $r \geq deg(u)$ since $\mathcal{S}$ is independent, so $deg(u) < \min\{d_u, r\}$ $(*)$. If $\{u, r\} \notin E(G)$, then apply the *Case* 1. Otherwise, there exists some vertex $v < r$ such that $\{v, u\} \notin E(G)$ since $deg(v) = d_v \geq d_u$ and $d_u > deg(u)$ by $(*)$. Since $deg(v) > deg(r)$, there exist a vertex w such that $\{v, w\} \in E(G)$ and $\{r, w\} \notin E(G)$. Then, remove the edge $\{v, w\}$ and add the edges $\{r, w\}$, $\{v, u\}$.

Case 4, if $\{1, 2, ..., r-1\} \in \mathcal{N}(r)$, and there exists some vertices $v, u \in V(G)$ such that $v < u < r$ and $\{v, u\} \notin E(G)$. Then, there exists vertices w and z such that $\{v, w\} \in E(G)$, $\{r, w\} \notin E(G)$ and $\{u, z\} \in E(G)$, $\{r, z\} \notin E(G)$ since $deg(v) \geq deg(u) > deg(r)$. If $w, z < r$, then *Case* 1 can be applied. If $w, z > r$, then switch the edges $\{v, w\}$, $\{u, z\}$ with the edges $\{v, u\}$, $\{w, r\}$.

If none of the four cases are satisfied, then the vertices $\{1, 2, ..., r\}$ are pairwise adjacent, and $deg(i) = \min\{d_i, r\}$ for every $i > r$. Since $\mathcal{S}$ is independent, then $\sum_{i=1}^{r} deg(i) = r(r-1) + \sum_{i=r+1}^{n} \min\{r, d_i\}$. By initial assumptions in (4.3), it can be also written that $\sum_{i=1}^{r} d_i \leq r(r-1) + \sum_{i=r+1}^{n} min(r, d_i)$, so $d_r - deg(r) \leq 0$ since $d_i = deg(i)$ for all $i < r$. Hence, the equation $d_r = deg(r)$ can be acquired by the inequalities $d_r - deg(r) \leq 0$ and $d_r - deg(r) \geq 0$. Now, the critical index value r can be increased by 1, and the same process can be repeated until a realization G is obtained where $deg(i) = d_i$ for all vertices $i \in V(G)$. $\square$

It is worth to note that the algorithm described in Figure 4.1 can produce the same realization in different edge sequence orders. For instance, the following edge sequences

$$E = (\{6,1\},\{6,3\},\{4,3\},\{4,1\},\{3,1\},\{3,5\},\{5,1\},\{5,2\},\{2,1\}),\ \textit{and}$$
$$\widetilde{E} = (\{6,3\},\{6,1\},\{4,3\},\{4,1\},\{3,1\},\{3,5\},\{5,1\},\{5,2\},\{2,1\})$$

represent the same graph, and can be generated for a given input sequence $\boldsymbol{d} = (5,2,4,2,3,2)$ with probabilities $1/56$, and $1/40$, respectively. Let $Graph(\cdot)$ be a function mapping every edge sequence to the corresponding graph, and let $\sigma(E)$ be the probability of generating the edge sequence E by the algorithm, so $Graph(E) = Graph(\widetilde{E})$, and $\sigma(E) = 1/56$, $\sigma(\widetilde{E}) = 1/40$.

Let $\mathcal{E}_{\boldsymbol{d}}$ be the set consisting of all possible edge sequences produced by the algorithm for the graphical degree sequence $\boldsymbol{d} = (d_1, ..., d_n)$. It can be said that two sequences $E, \widetilde{E} \in \mathcal{E}_{\boldsymbol{d}}$ are *equivalent* if $Graph(E) = Graph(\widetilde{E})$, and denoted by $E \equiv \widetilde{E}$. Therefore, the binary relation $\equiv$ defines an equivalence relation on the set $\mathcal{E}_{\boldsymbol{d}}$. Let $c(E)$ be the size of the equivalence class $[E]$ for the edge sequence $E \in \mathcal{E}_{\boldsymbol{d}}$, and let $\tilde{f}$ be the induced function over $\mathcal{E}_{\boldsymbol{d}}$ of the function f which is defined on the set $\mathcal{G}^{\boldsymbol{d}}$ such that $\tilde{f}(E) := f(Graph(E))$.

Proposition 4.2. *[1] Let π be a probability distribution on $\mathcal{G}^{\boldsymbol{d}}$. Then*

$$\mathbb{E}_{\sigma}\left(\frac{\tilde{f}(E)}{c(E)}\frac{d\tilde{\pi}}{d\sigma}(E)\right) = \mathbb{E}_{\pi} f(G) \tag{4.4}$$

where $\tilde{\pi}$ is absolutely continuous with respect to σ, and $\frac{d\tilde{\pi}}{d\sigma}$ is Radon–Nikodym derivative. Moreover, an unbiased estimator of $\mathbb{E}f(G)$ is

$$\widehat{f}_{\sigma,N} = \frac{1}{N}\sum_{i=1}^{N}\frac{\tilde{f}(E_i)}{c(E_i)}\frac{d\tilde{\pi}}{d\sigma}(E_i) \tag{4.5}$$

where E_1,...,E_N are the output edge sequences generated by independently running N times the algorithm in Figure 4.1 for the graphical degree sequence $\boldsymbol{d} = (d_1, ..., d_n)$.

Proof. Let E be an output sequence of edges produced by the algorithm.

$$\begin{aligned}
\mathbb{E}_\sigma\left(\frac{\tilde{f}(E)}{c(E)}\frac{d\tilde{\pi}}{d\sigma}(E)\right) &= \int_{\mathcal{E}_{\boldsymbol{d}}} \frac{\tilde{f}(\epsilon)}{c(\epsilon)}\frac{d\tilde{\pi}}{d\sigma}(\epsilon)\sigma(d\epsilon) = \int_{\mathcal{E}_{\boldsymbol{d}}} \frac{\tilde{f}(\epsilon)}{c(\epsilon)}\tilde{\pi}(d\epsilon) = \sum_{\epsilon\in\mathcal{E}_{\boldsymbol{d}}} \frac{\tilde{f}(\epsilon)}{c(\epsilon)}\tilde{\pi}(\epsilon) \\
&\overset{1}{=} \sum_{\zeta\in\mathcal{G}^{\boldsymbol{d}}} \sum_{\{\epsilon: Graph(\epsilon)=\zeta\}} \frac{\tilde{f}(\epsilon)}{c(\epsilon)}\tilde{\pi}(\epsilon) \\
&\overset{2}{=} \sum_{\zeta\in\mathcal{G}^{\boldsymbol{d}}} f(\zeta)\pi(\zeta) = \int_{\mathcal{G}_{\boldsymbol{d}}} f(\zeta)\pi(d\zeta) \\
&= \mathbb{E}_\pi f(G)
\end{aligned}$$

The sum in the equality (1) is computed over equivalence classes $[E] = \{\widetilde{E} \in \mathcal{E}_{\boldsymbol{d}} : Graph(E) = Graph(\widetilde{E})\}$, and the equality (2) follows from the fact that $c(E) = |\{\widetilde{E} \in \mathcal{E}_{\boldsymbol{d}} : Graph(E) = Graph(\widetilde{E})\}|$ since $c(E)$ is the size of the equivalence class $[E]$ and note that the induced functions $\tilde{f}(E)$, $\tilde{\pi}(E)$ are equal to $f(Graph(E))$ and $\pi(Graph(E))$ by the definition, respectively. □

Let $v_1, ..., v_k$ be the chosen indices in the second step of the algorithm in Figure 4.1, and $\boldsymbol{d} = \boldsymbol{d}^{(1)}, \boldsymbol{d}^{(2)}, ..., \boldsymbol{d}^{(k)}$ be the corresponding graphical degree sequences representing the degree sequence in the second step while choosing the indices. If two edge sequences are in the same equivalence class, it is clear that they must contain the same edges, possibly in a different order. An edge sequence can be divided into disjoint edge groups according to the chosen vertex v_i in the second step. Hence, each group in the sequence can have $d_{v_i}^{(i)}!$ different permutations, and the size of an equivalence class $[E]$ can be computed by $c(E) = \prod_{i=1}^{k} d_{v_i}^{(i)}!$.

Now, one can test any feature of the ensemble of simple graphs realizing a given degree sequence by using Proposition 4.2.

For instance, the cardinality of the set $\mathcal{G}^{\boldsymbol{d}}$ can be estimated by choosing the probability distribution π uniform over the measurable space $(\mathcal{G}^{\boldsymbol{d}}, \mathcal{P}(\mathcal{G}^{\boldsymbol{d}}))$, and letting $f(\cdot)$ be $1/|\mathcal{G}^{\boldsymbol{d}}|$. Then, the equation in Proposition 4.2 can be written as

$$|\mathcal{G}^d| = \mathbb{E}_\sigma \left(\frac{1}{c(E)} \frac{d\lambda}{d\sigma}(E) \right) \tag{4.6}$$

where $\lambda(\cdot)$ is the counting measure on $(\mathcal{E}_{\boldsymbol{d}}, \mathcal{P}(\mathcal{E}_{\boldsymbol{d}}))$. The number of simple graphs sharing the same graphical degree sequence with the friendship network in Figure 1.1 was estimated as $9.327 \pm 0.588 \times 10^{41}$ by using the unbiased estimator in (4.5) with 10.000 samples. The result is very close to the actual value, and the percent error is only %9.247.

4.1.2. Sequential Importance Sampling

In the importance sampling method described in the previous section, choosing a good proposal distribution for a given target measure can be problematic in some situations, especially, in high dimensional problems. In such cases, using sequential importance sampling technique can be more convenient. Let $\mu := \mu_1 \otimes \mu_2 \otimes \cdots \otimes \mu_d$ be a probability measure on $(X^d, \mathcal{X}^d)$ in which μ_1 is a probability measure over $(\mathsf{X}, \mathcal{X})$ and μ_i is a probability kernel from $(\mathsf{X}^{i-1}, \mathcal{X}^{i-1})$ to $(\mathsf{X}, \mathcal{X})$ for $2 \leq i \leq d$.

Assume that $\nu := \nu_1 \otimes \nu_2 \otimes \cdots \otimes \nu_d$ is the chosen proposal distribution and μ_i is absolutely continuous with respect to ν_i for each $1 \leq i \leq d$. The notation $\xi_{1:i}$ is used to denote the vector $(\xi_1, ..., \xi_i)$, and the expected value in (4.1) can be rewritten as

$$\mathbb{E}_\mu[f(\Xi_1, ..., \Xi_d)] = \int \cdots \int f(\xi_{1:d}) \frac{d\mu_1}{d\nu_1}(\xi_1) \left\{ \prod_{i=1}^{d-1} \frac{d\mu_{i+1}(\xi_{1:i}, \cdot)}{d\nu_{i+1}(\xi_{1:i}, \cdot)}(\xi_{i+1}) \right\} \nu(d\xi_{1:d}) \tag{4.7}$$

for a bounded measurable function f defined on X^d.

The Radon-Nikodym derivative is given by

$$w_d(\xi_{1:d}) := \frac{d\mu}{d\nu}(\xi_{1:d}) = \frac{d\mu_1}{d\nu_1}(\xi_1) \left\{ \prod_{i=1}^{d-1} \frac{d\mu_{i+1}(\xi_{1:i}, \cdot)}{d\nu_{i+1}(\xi_{1:i}, \cdot)}(\xi_{i+1}) \right\}$$

and the expectation in (4.7) can be approximated by the estimator

$$\widehat{f}_{\nu,N} = \frac{1}{N} \sum_{j=1}^{N} f(\xi_{1:d}^{(j)}) w_d(\xi_{1:d}^{(j)}) \tag{4.8}$$

where $\xi_{1:d}^{(j)} \sim \nu$ for $1 \leq j \leq N$. The *incremental weights* w_i can be recursively evaluated as follows:

$$w_1(\xi_1^{(j)}) = \frac{d\mu_1}{d\nu_1}(\xi_1^{(j)}), \quad w_{i+1}(\xi_{1:i+1}^{(j)}) := w_i(\xi_{1:i}^{(j)}) \frac{d\mu_{i+1}(\xi_{1:i}^{(j)}, \cdot)}{d\nu_{i+1}(\xi_{1:i}^{(j)}, \cdot)}(\xi_{i+1})$$

where $\xi_{1:i}^{(1)}, ..., \xi_{1:i}^{(N)}$ are samples from the proposal distribution ν_i for $1 \leq i < d$. Note that each sample $\xi_{1:d}^{(j)}$ can be also recursively constructed by drawing its each entry $\xi_i^{(j)}$ from the distribution $\nu_i(\xi_{i-1}^{(j)}, \cdot)$. The general schema of the sequential importance sampling technique is given in the algorithm of Figure 4.2 in order to generate samples $\xi_{1:d}^{(j)}$ and evaluate the weights w_i.

A matrix is called *binary* or *(0,1)-matrix* if its entries consist of only 0's or 1's. Let $\mathcal{M}^{(\boldsymbol{p},\boldsymbol{q})}$ be the set of $n \times m$ binary matrices with row sum $\boldsymbol{p} = (p_1, ..., p_n)$ and column sum $\boldsymbol{q} = (q_1, ..., q_m)$. Let μ be a uniform probability measure on the measurable space $(\mathcal{M}^{(\boldsymbol{p},\boldsymbol{q})}, \mathcal{P}(\mathcal{M}^{(\boldsymbol{p},\boldsymbol{q})})$. Since sampling directly from the distribution μ is very complicated due to the marginal sums, a more suitable probability measure can be considered as a proposal distribution to generate samples.

Output: N samples $\xi_{1:d}^{(j)} \sim \nu$ for $1 \leq j \leq N$, and weights w_i for $1 \leq i \leq d$

for $j = 1, ..., N$ **do**
  for $i = 1, ..., d$ **do**
    if $i = 1$ **then**
      Sample $\xi_1^{(j)} \sim \nu_1$
      $w_1 \leftarrow \frac{d\mu_1}{d\nu_1}(\xi_1^{(j)})$
    else
      Sample $\xi_i^{(j)} \sim \nu_i(\xi_{i-1}^{(j)}, \cdot)$
      $w_{i+1}(\xi_{1:i+1}^{(j)}) \leftarrow w_i(\xi_{1:i}^{(j)}) \frac{d\mu_{i+1}(\xi_{1:i}^{(j)}, \cdot)}{d\nu_{i+1}(\xi_{1:i}^{(j)}, \cdot)}(\xi_{i+1})$
    end if
  end for
end for

Figure 4.2. Sequential Importance Sampling Algorithm

Let $\boldsymbol{c}^{(1)}, ..., \boldsymbol{c}^{(m)}$ denote the possible column vectors of a $n \times m$ binary matrix with the column sum $\boldsymbol{q} = (q_1, ..., q_m)$. Note that the table can be constructed by sequentially sampling the binary vectors $\boldsymbol{c}^{(1)}, ..., \boldsymbol{c}^{(m)}$ with the sums $q_1, ..., q_m$, and updating the row sums at each step. In other words, the first column $\boldsymbol{c}^{(1)}$ can be sampled from a distribution, then the row sum $\boldsymbol{p}$ is updated to $\boldsymbol{p} - \boldsymbol{c}^{(1)}$. Similarly, the second and the other columns can be designed so that a (0,1)-matrix $M = (m_{ij})$ is obtained where $\sum_i m_{ij} = \boldsymbol{q}$.

Notice that some of the matrices constructed with this idea may not satisfy the row sum $\boldsymbol{p} = (p_1, ..., p_n)$. Hence, it is very important to choose a good proposal measure in order to reduce the number of invalid samples, and to increase the performance. Each column j with the sum q_j can be constructed by uniformly choosing q_j positions to place 1's, but this method is unfortunately very inefficient.

Chen *et al.* [2] proposed to select q_j positions with a probability proportionally to the corresponding row sums, and suggested to use *conditional-Poisson (CP) sampling* method with weights $w_i = \frac{p_i^{(j-1)}}{m-j+1-p_i^{(j-1)}}$ where $p_i^{(j-1)}$ is the ith row sum after sampling the first $j-1$ columns.

Let $R_1, ..., R_n$ be random variables where $R_i \sim \text{Bern}(\pi_i)$ and $S_R = R_1 + \cdots + R_n$. Then the random variable S_R follows *Poisson-binomial distribution*, or follows Binomial distribution if all the probabilities π_i are the same. Let the set $\mathcal{I}$ be $\{1, ..., n\}$, and the conditional distribution of $R = (R_1, ..., R_n)$ given S_R is called *CP distribution* and

$$\begin{aligned}\mathbb{P}(R_1 = r_1, ..., R_n = r_n | S_R = k) &= \frac{\mathbb{P}(R_1 = r_1, ..., R_n = r_n, S_R = k)}{\mathbb{P}(S_R = k)} \\ &= \frac{\prod_{i=1}^{n} \pi_i^{r_i}(1-\pi_i)^{1-r_i}}{\sum_{\substack{\boldsymbol{r}\in\{0,1\}^n \\ \sum_j r_j = k}} \prod_{i=1}^{n} \pi_i^{r_i}(1-\pi_i)^{1-r_i}} = \frac{\prod_{i=1}^{n} w_i^{r_i}(1-\pi_i)}{\sum_{\substack{\boldsymbol{r}\in\{0,1\}^n \\ \sum_j r_j = k}} \prod_{i=1}^{n} w_i^{r_i}(1-\pi_i)} \\ &= \frac{\prod_{i=1}^{n} w_i^{r_i}}{\sum_{\substack{A\subseteq\mathcal{I} \\ |A|=k}} \prod_{i\in A} w_i} = \frac{\prod_{i=1}^{n} w_i^{r_i}}{R(k, \mathcal{I})}\end{aligned}$$

where $R(k, S) := \sum_{\substack{A\subseteq S \\ |A|=k}} \prod_{i\in A} w_i$, and w_i are defined by $\frac{\pi_i}{1-\pi_i}$. Note that $R(0, S) = 0$, $R(k, S) = 0$ for $k > |S|$, and $\mathbb{P}$ is the probability measure on $(\{0,1\}^n, \mathcal{P}(\{0,1\}^n))$

It can be seen that the marginal distribution of the column j after selecting the columns $\boldsymbol{c}^{(1)}, ..., \boldsymbol{c}^{(j-1)}$ with updated row sum $\boldsymbol{p}^{(j-1)}$ is the same as the conditional distribution of $R = (R_1, ..., R_n)$ given $S_R = q_j$ where $R_i \sim \text{Bern}(\pi_i)$, and $\pi_i = \frac{p_i^{(j-1)}}{m-(j-1)}$ [2]. Hence, taking CP distribution as proposal distribution is a good choice. Chen and Liu [35] proposed five different methods to draw samples from the CP distribution. The *drafting sampling* procedure is preferred to apply for this problem, and the idea behind the method is as follows: Let S_i be the set of selected indices after the ith step, so $S_0 = \emptyset$ and $|S_i| = i$ for each $0 \leq i \leq n$.

A new element k from the set $\mathcal{I}\backslash S_{i-1}$ is chosen with probability $\mathbb{P}(i, \mathcal{I}\backslash S_{i-1})$ which is equal to

$$\mathbb{P}(k, \mathcal{I}\backslash S_{i-1}) = \frac{w_k R(c-k, \mathcal{I} \cup (S_{i-1} \cup \{k\}))}{(c-i-1)R(c-i+1, \mathcal{I}\backslash S_{i-1})} \tag{4.9}$$

where the weights $w_k = \frac{\pi_k}{1-\pi_k}$. The value of the function $R(k, S)$ can be more effectively computed in a recursive way by using the method suggested by Gail, Lubin and Rubinstein [36], which is given by $R(k, S) = R(k, S\backslash\{k\}) + w_k R(k-1, S\backslash\{k\})$

Hence, a binary matrix can be generated by sequentially sampling the column vectors from the conditional Poisson distributions ν_i as described above. Let X^i be the set of all binary column vectors with sums $(q_1, ..., q_i)$, and ν be the probability distribution on $(\mathsf{X}^m, \mathcal{X}^m)$ which is defined by $\nu_1 \otimes \nu_2 \cdots \otimes \nu_m$ where ν_1 is a measure on $(\mathsf{X}, \mathcal{X})$, and ν_i is a transition kernel from $(\mathsf{X}^{i-1}, \mathcal{X}^{i-1})$ to $(\mathsf{X}, \mathcal{X})$. Then, ν can be considered in sequential importance sampling method as a proposal distribution for the computation of expected values related to graph properties. For instance, the expectation of Radon–Nikodym derivative $\frac{d\lambda}{d\nu}$ with respect to the measure ν gives the cardinality of the set $\mathcal{M}^{(\boldsymbol{p},\boldsymbol{q})}$.

$$\mathbb{E}_\nu \left[\frac{d\lambda}{d\nu}(\Xi) \right] = \int_{\mathsf{X}^m} \frac{d\lambda}{d\nu}(\xi) d\nu = \int_{\mathsf{X}^m} d\lambda = \left| \mathcal{M}^{(\boldsymbol{p},\boldsymbol{q})} \right| \tag{4.10}$$

where $\lambda(\cdot)$ is the counting measure, and it is absolutely continuous with respect to ν since $\nu(\xi) > 0$ for any $\xi \in \mathsf{X}^d$. The algorithm is tested on a small example, and the number of binary matrices satisfying the column and row sums $(2, 2, 2, 2, 2, 2)$ is estimated to be $(65,505 \pm 0,448) \times 10^3$ with 1000 samples while the actual result is equal to 67.950.

4.2. Uniform Sampling Methods

The Monte Carlo methods presented in the previous section uses alternative distributions to construct graph samples so choosing a good proposal distribution is of great importance for the performance of the algorithms. This section aims to directly generate uniformly distributed realizations of a given degree sequence. A uniform sampling algorithm for binary matrices with margin sums (or equivalently bipartite graphs) is investigated, and a new uniform sampling method for simple graphs is developed based on this work.

4.2.1. Bipartite Graphs

Let $N(\boldsymbol{p}, \boldsymbol{q})$ be the number of *(0,1)-matrices* where the row and column sums are $\boldsymbol{p} = (p_1, ..., p_n) \in N^n$, $\boldsymbol{q} = (q_1, ..., q_m) \in \mathbb{N}^m$, respectively. It is clear that the sums of the sequences must be equal to each other in order to find a valid binary matrix satisfying the given margin sums. However, this condition alone is not sufficient to ensure the existence of a binary matrix meeting with the margin sums $(\boldsymbol{p}, \boldsymbol{q})$. In 1957, Gale and Ryser suggested a theorem which provides necessary and sufficient conditions for (0,1)-matrices with prescribed row and column sums. It is analogous to the Erdös-Gallai Theorem 4.1 proposed for simple graphs, and it will be used in the implementation of the sampling algorithm presented in this section.

Theorem 4.3. *(Gale-Ryser) [37] Let* $(\boldsymbol{p}, \boldsymbol{q}) \in \mathbb{N}^n \times \mathbb{N}^m$ *be a pair of non-increasing sequences such that* $\sum_{i=1}^{n} p_i = \sum_{j=1}^{m} q_j$. *It is said that* $\boldsymbol{p}$ *is dominated by* $\boldsymbol{q}^*$, *and denoted by* $\boldsymbol{p} \trianglelefteq \boldsymbol{q}^*$ *if* $\sum_{i=1}^{k} p_i \leq \sum_{j=1}^{k} q_i^*$ *for all* k *where* $1 \leq k \leq \max\{n, m\}$, $p_i := 0$ *for* $i > m$, $q_j^* := 0$ *for* $j > n$ *and* $q_j^* := |q_l : q_l \geq j, \quad l \in \{1, ..., m\}|$.
Then, there exists a (0-1)-matrix $\boldsymbol{A} = (a_{ij})$ *where* $\sum_{j=1}^{m} a_{ij} = \boldsymbol{p}$, $\sum_{i=1}^{n} a_{ij} = \boldsymbol{q}$ *if and only if* $\boldsymbol{p}$ *is dominated by* $\boldsymbol{q}^*$.

Proof. Let $r(A)$ be the row sum, $c(A)$ be the column sum of a given binary matrix A, and $k := \max\{n, m\}$.

For the necessity part, let $A = (a_{ij})$ be any (0-1)-matrix satisfying $r(A) = \boldsymbol{p}$ and $c(A) = \boldsymbol{q}$. If A does not contain any *gap*, in other words, if it does not have any entry $a_{uv} = 0$ such that $a_{iv} = 1$ where $i > u$. Then $\boldsymbol{q}^* = \boldsymbol{p}$ since the sequences $\boldsymbol{p}$ is non-increasing, so $\boldsymbol{q}^* \trianglerighteq \boldsymbol{q}^* = \boldsymbol{p}$. If the matrix A contains a gap, then there exists entries $a_{uv} = 0$ and $a_{\xi v} = 1$ for some $\xi > u$. If the values of a_{uv} and $a_{\xi v}$ are switched, then a new matrix $\widetilde{A} = (\tilde{a}_{ij})$ is obtained by closing the gap without changing the row sum $c(A) = c(\widetilde{A})$. Since $\sum_{i=1}^{l} \tilde{a}_{ij} \geq \sum_{i=1}^{l} a_{ij}$ for $l < \xi$ and $\sum_{i=1}^{l} \tilde{a}_{ij} = \sum_{i=1}^{l} a_{ij}$ for $l \geq \xi$, it can be written that $r(\widetilde{A}) \trianglerighteq r(A)$. Therefore, if this process is repeated for each column until no gap remains, then a new binary matrix $\widetilde{A}$ is obtained such that $\boldsymbol{q}^* = r(\widetilde{A})$ so $\boldsymbol{q}^* \trianglerighteq \boldsymbol{p}$ since $\boldsymbol{q}^* = r(\widetilde{A}) \trianglerighteq r(A) = \boldsymbol{p}$.

For sufficiency, assume that $(\boldsymbol{p}, \boldsymbol{q})$ are partitions such that $\sum_{i=1}^{k} p_i = \sum_{j=1}^{k} q_j$, and $\boldsymbol{q}^* \trianglerighteq \boldsymbol{p}$. A matrix $A = (a_{ij})$ without any gap and $c(A) = \boldsymbol{q}$ can be easily constructed, and let A be such a matrix. Then, $r(A) \trianglerighteq \boldsymbol{p}$ since $r(A) = \boldsymbol{q}^*$ and $\boldsymbol{q}^* \trianglerighteq \boldsymbol{p}$. If $r(A) = \boldsymbol{p}$, then the proof completes. Therefore, suppose that $r(A) = (r_1, r_2, ..., r_n) \neq \boldsymbol{p}$. Let u and ξ be minimal indices such that $r_u > p_u$ and $r_\xi < p_\xi$. Since $r(A) \trianglerighteq \boldsymbol{p}$, u must be less than ξ. Then, a column index v can be found such that $a_{uv} = 1$, and $a_{\xi v} = 0$. By switching the values of the entries a_{uv} and $a_{\xi v}$, a new matrix $\widetilde{A}$ with row sum $r(\widetilde{A}) = (r_1, ..., r_{u-1}, r_u - 1, r_{u+1}, ..., r_{\xi-1}, r_\xi + 1, r_{\xi+1}, ..., r_n)$ is obtained. Note that $\| \ r(\widetilde{A}) - \boldsymbol{p} \ \|_2 < \| \ r(A) - \boldsymbol{p} \ \|_2$ and $r(\widetilde{A}) \trianglerighteq \boldsymbol{p}$. Hence, if this argument is proceeded until $r(\widetilde{A}) = \boldsymbol{p}$, then the desired matrix $\widetilde{A} = (\widetilde{a}_{ij})$ is attained where $\sum_{j=1}^{m} \widetilde{a}_{ij} = \boldsymbol{p}$, $\sum_{i=1}^{n} \widetilde{a}_{ij} = \boldsymbol{q}$. □

For a given row and column sums $(\boldsymbol{p}, \boldsymbol{q}) \in \mathbb{N}^n \times \mathbb{N}^m$, a binary matrix satisfying the margin sums $\boldsymbol{p}$ and $\boldsymbol{q}$ can be built by successively filling the rows of an initially empty table, starting from the first row up to the last row. Consider the example in Figure 4.3 which illustrates the construction of a binary matrix. The first row of Table 1 is chosen as $\boldsymbol{u} = (1, 1, 0, 1, 1)$, and then the row and column sums are updated to $\mathcal{L}\boldsymbol{p} = (3, 2)$, and $\boldsymbol{q} - \boldsymbol{u} = (2, 1, 1, 1, 0)$ where $\mathcal{L}$ is the left shift operator. The same procedure can be recursively applied to the remaining unfilled part of the table with the updated margin sums $(\mathcal{L}\boldsymbol{p}, \boldsymbol{q} - \boldsymbol{u})$ until it is completely filled with 1's and 0's.

	3	2	1	2	1
4					
3					
2					

Table 1

	3	2	1	2	1
4	1	1	0	1	1
3					
2					

Table 2

	3	2	1	2	1
4	1	1	0	1	1
3	1	1	0	1	0
2					

Table 3

	3	2	1	2	1
4	1	1	0	1	1
3	1	1	0	1	0
2	1	0	1	0	0

Table 4

Figure 4.3. Construction of a Binary Matrix with Prescribed Column and Row Sums

Let $b = \max_j q_j$, and the columns of a table can be divided into b disjoint groups where each group k consists of the columns whose sum equal to k. For instance, the groups of Table 1 in Figure 4.3 are represented by different colors. Note that in a row vector $\boldsymbol{u}$, changing the positions of 1's in the same groups do not affect the possible number of matrices for the remaining unfilled part of the table. Hence, the values of the functions $N(\mathcal{L}\boldsymbol{p}, \boldsymbol{q}-\boldsymbol{u})$ and $N(\mathcal{L}\boldsymbol{p}, \boldsymbol{q}-\boldsymbol{v})$ are equal to each other for any permutation $\boldsymbol{v}$ of the vector $\boldsymbol{u}$ in which the number of $1's$ in each group remains unchanged.

Let $\bar{\boldsymbol{q}} = (\bar{q}_1, ..., \bar{q}_m)$ be a vector such that $\bar{q}_k = |\{q_j : q_j = k,\ 1 \leq j \leq m\}|$. Notice that each element $\bar{q}_k$ of the vector $\bar{\boldsymbol{q}}$ gives the number of columns whose sum equal to k, which is size of the group k. In the construction of the first row of the table in Figure 4.3, one position from the first and the third groups, and two positions from the second group are chosen. Note that $\bar{\boldsymbol{q}} = (2, 2, 1)$ and there are $\binom{2}{1}\binom{2}{2}\binom{1}{1}$ permutation vectors $\boldsymbol{v}$ of the first row vector $\boldsymbol{u}$ where $N(\mathcal{L}\boldsymbol{p}, \boldsymbol{q}-\boldsymbol{u}) = N(\mathcal{L}\boldsymbol{p}, \boldsymbol{q}-\boldsymbol{v})$. In general, the number of permutations of a row vector $\boldsymbol{u}$ can be computed by $\binom{\boldsymbol{q}}{\boldsymbol{s}} := \binom{q_1}{s_1} \cdots \binom{q_b}{s_b}$ where s_k is the number of elements u_j where $q_j = k$, and the proof is given in Lemma 4.5.

Let $\boldsymbol{q} = (q_1, ..., q_m) \in \mathbb{N}^m$ and $\boldsymbol{r} = (r_1, ..., r_m) \in \mathbb{N}^m$ be two finite sequences of length m and it is denoted by $\boldsymbol{r} \leq \boldsymbol{q}$, if $r_j \leq q_j$ for all $j \in \{1, ..., m\}$. The binary operator "$\backslash$" is defined by $\boldsymbol{q}\backslash\boldsymbol{r} := \boldsymbol{q} - \boldsymbol{r} + \mathcal{L}\boldsymbol{r}$, and the operator $\wedge$ is given by $\boldsymbol{q} \wedge \boldsymbol{r} := (\min(q_1, r_1), ..., \min(q_m, r_m))$.

The following lemma proves that there always exists a corresponding vector $\boldsymbol{s}$ for a given vector $\boldsymbol{r} \in \mathbb{N}^m$ and for a chosen binary row vector $\boldsymbol{u} \in \{0,1\}^n$ satisfying $\boldsymbol{r} \geq \boldsymbol{u}$.

Lemma 4.4. *[3] Let $\boldsymbol{r} \in \mathbb{N}^n$, and $\boldsymbol{u} \in \{0,1\}^n$ such that $\boldsymbol{r} \geq \boldsymbol{u}$. Then $\overline{\boldsymbol{r}-\boldsymbol{u}} = \overline{\boldsymbol{r}} \backslash \boldsymbol{s}$ if and only if $\boldsymbol{s} = \boldsymbol{r}^* - (\boldsymbol{r}-\boldsymbol{u})^*$*

Proof. Let $\mathcal{I}$ be an identity operator, and $\boldsymbol{r} \in \mathbb{N}^n$. Observe that $(\sum_{i=0}^{n} \mathcal{L}^i)\overline{\boldsymbol{r}} = \boldsymbol{r}^*$, and $\overline{\boldsymbol{r}} = \boldsymbol{r}^* - \mathcal{L}\boldsymbol{r}^* = (\mathcal{I}-\mathcal{L})\boldsymbol{r}^*$. It can be also written that $(\sum_{i=0}^{n} \mathcal{L}^i)(\mathcal{I}-\mathcal{L})\boldsymbol{r} = (\mathcal{I}-\mathcal{L})(\sum_{i=0}^{n} \mathcal{L}^i)\boldsymbol{r} = \boldsymbol{r}$, so $(\mathcal{I}-\mathcal{L})^{-1} = (\sum_{i=0}^{n} \mathcal{L}^i)$. Hence $\overline{\boldsymbol{r}-\boldsymbol{u}} = \overline{\boldsymbol{r}} \backslash \boldsymbol{s} = \overline{\boldsymbol{r}} - \boldsymbol{s} + \mathcal{L}\boldsymbol{s} = \overline{\boldsymbol{r}} - (\mathcal{I}-\mathcal{L})\boldsymbol{s}$ if and only if $(\mathcal{I}-\mathcal{L})\boldsymbol{s} = \overline{\boldsymbol{r}} - \overline{\boldsymbol{r}-\boldsymbol{u}}$ if and only if $\boldsymbol{s} = (\mathcal{I}-\mathcal{L})^{-1}(\overline{\boldsymbol{r}} - \overline{\boldsymbol{r}-\boldsymbol{u}}) = (\sum_{i=0}^{n} \mathcal{L}^i)(\overline{\boldsymbol{r}} - \overline{\boldsymbol{r}-\boldsymbol{u}}) = \boldsymbol{r}^* - (\boldsymbol{r}-\boldsymbol{u})^*$ □

Let $\mathcal{C}_b(k) := \{\boldsymbol{s} \in \mathbb{N}^b : \sum_{j=1} s_j = k\}$, and $\mathcal{C}^{\boldsymbol{r}}(k) := \{\boldsymbol{s} \in \mathcal{C}_b(k) : \boldsymbol{s} \leq \boldsymbol{r}\}$. For any binary vector $\boldsymbol{u} \in \{0,1\}^m$ with the column sum $\boldsymbol{q} \in \mathbb{N}^m$, there exists a unique sequence $\boldsymbol{s} = (s_1, ..., s_b) \in \mathbb{N}^b$ satisfying the equality $\overline{\boldsymbol{q}-\boldsymbol{u}} = \overline{\boldsymbol{q}} \backslash \boldsymbol{s}$ by Lemma 4.4. It is clear that a vector $\boldsymbol{v}$ is a permutation of $\boldsymbol{u}$ as described above whenever $\overline{\boldsymbol{q}-\boldsymbol{u}} = \overline{\boldsymbol{q}-\boldsymbol{v}}$. Hence, the set $\mathcal{P}_{\boldsymbol{s}} := \{\boldsymbol{v} \in \{0,1\}^n : \boldsymbol{v} \leq \boldsymbol{q},\ \overline{\boldsymbol{q}-\boldsymbol{v}} = \overline{\boldsymbol{q}} \backslash \boldsymbol{s}\}$ contains all the permutations of the vector $\boldsymbol{u}$. Then, the cardinality of the set $\mathcal{P}_{\boldsymbol{s}}$ is $\binom{\overline{\boldsymbol{q}}}{\boldsymbol{s}}$ by the following Lemma 4.5 where $\binom{\overline{\boldsymbol{q}}}{\boldsymbol{s}} := \binom{\overline{q}_1}{s_1} \cdots \binom{\overline{q}_b}{s_b}$

Lemma 4.5. *[3] Let $\boldsymbol{q} \in \mathbb{N}^m$, and $\boldsymbol{s} \in \mathbb{N}^b$ such that $\boldsymbol{s} \leqslant \overline{\boldsymbol{q}}$, and $b := \max_j q_j$. Let $k = \sum_i s_i$, then*

$$\left| \{\boldsymbol{u} \in \mathcal{C}^{q \wedge 1}(k) : \overline{\boldsymbol{q}-\boldsymbol{u}} = \overline{\boldsymbol{q}} \backslash \boldsymbol{s}\} \right| = \binom{\overline{\boldsymbol{q}}}{\boldsymbol{s}} \tag{4.11}$$

Proof. Note that the sequence $\boldsymbol{q} = (q_1, .., q_m)$ contains $\overline{q}_j$ number of terms equal to j. For each term s_j of the sequence $\boldsymbol{s} = (s_1, ..., s_b)$, one can choose s_j positions out of the $\overline{q}_j$ possible coordinates where $q_i = j$ in the construction of a binary vector $\boldsymbol{u} \in \mathcal{C}^{q \wedge 1}(k)$. Hence, one can find $\binom{\overline{q}_j}{s_j}$ possible ways to place $1's$ for the j'th element of the sequence $\boldsymbol{s}$. If the same argument is repeated for all $1 \leq j \leq b$, then it can be found that there are $\binom{\overline{q}_1}{s_1}\binom{\overline{q}_2}{s_2}...\binom{\overline{q}_b}{s_b} = \binom{\overline{\boldsymbol{q}}}{\boldsymbol{s}}$ possible ways to construct the vector $\boldsymbol{u}$ satisfying $\overline{\boldsymbol{q}-\boldsymbol{u}} = \overline{\boldsymbol{q}} \backslash \boldsymbol{s}$. □

Let the function $\overline{N}(\boldsymbol{p}, \overline{\boldsymbol{q}})$ be $N(\boldsymbol{p}, \boldsymbol{q})$ for a given pair of sequences $(\boldsymbol{p}, \boldsymbol{q}) \in \mathbb{N}^n \times \mathbb{N}^m$, and $\overline{N}(\boldsymbol{p}, \overline{\boldsymbol{q}})$ can be written in a recursive form by the following Theorem 4.6.

Theorem 4.6. *[3] The number of (0-1)-matrices with row and column sums $\boldsymbol{p} \in \mathbb{N}^n$, $\boldsymbol{q} \in \mathbb{N}^m$ is given by*

$$\overline{N}(\boldsymbol{p}, \overline{\boldsymbol{q}}) = \sum_{\boldsymbol{s} \in C^{\overline{q}}(p_1)} \binom{\overline{\boldsymbol{q}}}{\boldsymbol{s}} \overline{N}(\mathcal{L}\boldsymbol{p}, \overline{\boldsymbol{q}} \backslash \boldsymbol{s}) \tag{4.12}$$

Proof. Let $(\boldsymbol{p}, \boldsymbol{q}) \in \mathbb{N}^n \times \mathbb{N}^m$. Then

$$\begin{aligned} \overline{N}(\boldsymbol{p}, \overline{\boldsymbol{q}}) := N(\boldsymbol{p}, \boldsymbol{q}) &\overset{1}{=} \sum_{\boldsymbol{u} \in C^{q \wedge 1}(p_1)} N(\mathcal{L}\boldsymbol{p}, \boldsymbol{q} - \boldsymbol{u}) \\ &\overset{2}{=} \sum_{\boldsymbol{s} \in C^{\overline{q}}(p_1)} \sum_{\substack{\boldsymbol{u} \in C^{q \wedge 1}(p_1) \\ \overline{q - u} = \overline{q} \backslash s}} N(\mathcal{L}\boldsymbol{p}, \boldsymbol{q} - \boldsymbol{u}) \\ &\overset{3}{=} \sum_{\boldsymbol{s} \in C^{\overline{q}}(p_1)} \binom{\overline{\boldsymbol{q}}}{\boldsymbol{s}} \overline{N}(\mathcal{L}\boldsymbol{p}, \overline{\boldsymbol{q}} \backslash \boldsymbol{s}) \end{aligned}$$

In the first equality, p_1 possible places are chosen to construct the first row of a binary matrix. The second step divides the sum in the previous equation into disjoint classes and one can always find a vector $\boldsymbol{s} \in C^{\overline{\boldsymbol{q}}}(p_1)$ satisfying $\overline{\boldsymbol{q} - \boldsymbol{u}} = \overline{\boldsymbol{q}} \backslash \boldsymbol{s}$ by Lemma 4.4 for any $\boldsymbol{u} \in C^{q \wedge 1}(p_1)$. Finally, the last equality follows from Lemma 4.5, and from the fact that if $\overline{(\boldsymbol{q} - \boldsymbol{u})} = \overline{\boldsymbol{q}} \backslash \boldsymbol{s}$ then $N(L\boldsymbol{p}, \boldsymbol{q} - \boldsymbol{u}) = \overline{N}(L\boldsymbol{p}, \overline{\boldsymbol{q} - \boldsymbol{u}}) = \overline{N}(L\boldsymbol{p}, \overline{\boldsymbol{q}} \backslash \boldsymbol{s})$. □

Miller and Harrison [3] suggested an enumeration algorithm based on Theorem 4.6 for (0,1)-matrices satisfying given margin sums, or equivalently, for bipartite graphs with prescribed degree sequence. The details of the method are given in the algorithm of Figure 4.4 below.

Input: A pair $(\boldsymbol{p}, \overline{\boldsymbol{q}})$ where $(\boldsymbol{p}, \boldsymbol{q}) \in \mathbb{N}^n \times \mathbb{N}^m$ and $\sum_{i=1}^{n} p_i = \sum_{j=1}^{m} q_j$

Output: The number of bipartite graphs realizing the input pair $(\boldsymbol{p}, \boldsymbol{q})$

Storage: Lookup table storing intermediate results, initialized with $\overline{\mathcal{N}}(0,0) = 1$

1: Check the lookup table, if $\overline{\mathcal{N}}(\boldsymbol{p}, \overline{\boldsymbol{q}})$ was cached before, return it.

2: If there exists no binary matrix realizing the inputs, then cache the result, and return 0

3: For each possible sequence $\boldsymbol{s} \in \mathcal{C}^{\overline{\boldsymbol{q}}}(p_1)$, compute $\mathcal{N}(\mathcal{L}\boldsymbol{p}, \overline{\boldsymbol{q}} \backslash \boldsymbol{s})$

4: Evaluate the sum $\sum_{\boldsymbol{s} \in \mathcal{C}^{\overline{\boldsymbol{q}}}(p_1)} \binom{\overline{\boldsymbol{q}}}{\boldsymbol{s}} \mathcal{N}(\mathcal{L}\boldsymbol{p}, \overline{\boldsymbol{q}} \backslash \boldsymbol{s})$, and cache the result.

5: Return the result found in the previous step (4).

Figure 4.4. Enumeration of Bipartite Graphs with Prescribed Degree Sequences

Note that the fourth step of the algorithm can be factorized as follows:

$$\begin{aligned}\sum_{s \in C^{\overline{q}}(p_1)} \binom{\overline{\boldsymbol{q}}}{\boldsymbol{s}} \mathcal{N}(\mathcal{L}p, \overline{\boldsymbol{q}} \backslash \boldsymbol{s}) &= \sum_{s \in C^{\overline{q}}(p_1)} \binom{\overline{q}_b}{s_b} \cdots \binom{\overline{q}_2}{s_2} \binom{\overline{q}_1}{s_1} \mathcal{N}(\mathcal{L}p, \overline{\boldsymbol{q}} \backslash \boldsymbol{s}) \\ &= \sum_{s_b} \binom{\overline{q}_b}{s_b} \cdots \sum_{s_2} \binom{\overline{q}_2}{s_2} \sum_{s_1} \binom{\overline{q}_1}{s_1} \mathcal{N}(\mathcal{L}p, \overline{\boldsymbol{q}} \backslash \boldsymbol{s})\end{aligned}$$

Instead of checking all sequences $\boldsymbol{s} = (s_1, ...s_m)$ in the set $\mathcal{C}^{\overline{q}}(p_1)$, one can consider only suitable sequences satisfying the Gale-Ryser conditions by combining the third and fourth steps of the algorithm. Let j be the column index of a table, $\tilde{l}$ be the lower bound, and $\tilde{u}$ be the upper bound for a set of values. After choosing the first $j-1$ elements of the sequence $\boldsymbol{s} = (s_1, ..., s_{j-1}, \boxdot, ..., \boxdot)$, the element s_j can be picked from the set $S_j := \{k \in \mathbb{N} : \tilde{l} \leq k \leq \tilde{u}\}$ where $\tilde{l} := \max\{0,\ p_1 - \sum_{k=1}^{j-1} s_k - r^*_{j+1}\}$, $\tilde{u} := min\{\boldsymbol{r}_j,\ p_1 - \sum_{k=1}^{j-1} s_k,\ \widetilde{gr}\}$, $\widetilde{gr} := \sum_{k=1}^{j} r^*_k - \sum_{k=2}^{j+1} p_k$, and $\boldsymbol{r} = \overline{\boldsymbol{q}} \backslash \big(s_1, ..., s_{j-1}, 0, .., 0\big)$. For each choice of s_j, the Gale-Ryser condition, $\widetilde{gr}$, guarantees the existence of a binary matrix for the unfilled part.

	2	1	1
2			
2			

Table 1

	2	1	1
2	0	1	1
2			

Table 2

	2	1	1
2	1	1	0
2			

Table 3

	2	1	1
2	1	1	0
2	1	0	1

Table 4

Figure 4.5. Construction of a Binary Matrix with Gale-Ryser Conditions

For example, let $\boldsymbol{p} = (2, 2)$ and $\boldsymbol{q} = (2, 1, 1)$, so the vector of occurrences $\overline{\boldsymbol{q}} = (2, 1)$ and $\boldsymbol{q}^* = (3, 1)$ (Table 1 in Figure 4.5). For $j = 1$, $s_1 \in \{1\}$ since $\tilde{l} = \max\{0, 1\} = 1$ and $\tilde{u} = \min\{3, 2, 1\} = 1$, similarly, for the second column, $s_2 \in \{1\}$ because $\tilde{l} = \max\{0, 1\} = 1$ and $\tilde{u} = \min\{1, 1, 1\} = 1$. Therefore, $\boldsymbol{s}$ becomes $(1, 1)$ for the first row. If the same argument is proceeded, one can find $\boldsymbol{s} = (2, 0)$ for the second row. Thus, there exists $\binom{\overline{\boldsymbol{q}}}{\boldsymbol{s}} = \binom{2}{1}\binom{1}{1} = 2$ and $\binom{(2,1)\backslash(1,1)}{\boldsymbol{s}} = \binom{2}{2}\binom{0}{0} = 1$ possible choices for the first and the second rows, respectively.

The construction of a valid (0,1)-matrix realizing the margin $(\boldsymbol{p}, \boldsymbol{q})$ is illustrated in Table 3 and in Table 4 of Figure 4.5. However, if Gale-Ryser conditions are not taken into consideration, then one can choose $\boldsymbol{s}$ to be $(2, 0) \in \mathcal{C}^{\overline{q}}(2)$ for the first row so the first row must be $(0, 1, 1)$ but a valid binary vector for the second row cannot be found (Table 2).

Notice that the algorithm given in Figure 4.4 traverses the pairs of vectors $(\boldsymbol{u}, \overline{\boldsymbol{v}})$ by initially starting with $(\boldsymbol{p}, \boldsymbol{q})$, and the set of all traversed pairs $(\boldsymbol{u}, \overline{\boldsymbol{v}})$ form a tree with a root node $(\boldsymbol{p}, \boldsymbol{q})$. Let $\mathcal{D}(\boldsymbol{p}, \boldsymbol{q})$ denote the set of pairs $(\boldsymbol{u}, \overline{\boldsymbol{v}})$ traversed by the algorithm for a given initial input pair $(\boldsymbol{p}, \overline{\boldsymbol{q}})$ where $(\boldsymbol{p}, \boldsymbol{q}) \in \mathbb{N}^n \times \mathbb{N}^m$. A pair $(\boldsymbol{\xi}, \overline{\boldsymbol{\eta}})$ is said to be a child of $(\boldsymbol{u}, \overline{\boldsymbol{v}})$ if there exists $\boldsymbol{s} \in \mathcal{C}^{\overline{v}}(u_1)$ such that $\overline{\boldsymbol{\eta}} = \overline{\boldsymbol{v}}\backslash\boldsymbol{s}$ and $\mathcal{L}\boldsymbol{u} = \boldsymbol{\xi}$. With this observation, the uniform sampling algorithm from the set of (0,1)-matrices realizing the given margins $(\boldsymbol{p}, \boldsymbol{q})$ can be presented.

Input: A pair $(\boldsymbol{p}, \overline{\boldsymbol{q}})$ where $(\boldsymbol{p}, \boldsymbol{q}) \in \mathbb{N}^n \times \mathbb{N}^m$ and $\sum_{i=1}^{n} p_i = \sum_{j=1}^{m} q_j$

Output: A uniform sample from the set of (0,1)-matrices with margin sums $(\boldsymbol{p}, \boldsymbol{q})$

Storage: Lookup table storing the number of matrices realizing the margins $(\boldsymbol{u}, \boldsymbol{v})$ for each pair $(\boldsymbol{u}, \overline{\boldsymbol{v}}) \in \mathcal{D}(p, q)$

1: Initialize $(\boldsymbol{u}, \boldsymbol{v})$ with $(\boldsymbol{p}, \boldsymbol{q})$

2: If $(\boldsymbol{u}, \boldsymbol{v}) = (\boldsymbol{0}, \boldsymbol{0})$, terminate with the set of constructed row vectors.

3: Select a child $(\mathcal{L}\boldsymbol{u}, \overline{\boldsymbol{v}} \backslash \boldsymbol{s}) \in \mathcal{D}(\boldsymbol{p}, \overline{\boldsymbol{q}})$ of $(\boldsymbol{u}, \overline{\boldsymbol{v}})$ with a probability proportional to its count times the number of vectors $\boldsymbol{r} \in \mathcal{C}^{\boldsymbol{v}^{\wedge 1}}(u_1)$ such that $\overline{\boldsymbol{v} - \boldsymbol{r}} = \overline{\boldsymbol{v}} \backslash \boldsymbol{s}$.

4: Uniformly choose a row $\boldsymbol{r} \in \mathcal{C}^{\boldsymbol{v}^{\wedge 1}}(u_1)$ from the set $\{\xi : \overline{\boldsymbol{v} - \boldsymbol{\xi}} = \overline{\boldsymbol{v}} \backslash \boldsymbol{s}\}$ at random.

5: Set $(\boldsymbol{u}, \boldsymbol{v})$ to $(\mathcal{L}\boldsymbol{u}, \boldsymbol{v} - \boldsymbol{r})$.

6: Return to the step (2)

Figure 4.6. Uniform Sampling of Bipartite Graphs with Prescribed Degree Sequences

Miller and Harrison [3] showed that the computation time of the counting Algorithm 4.4 is in $\mathcal{O}\big(nm^b(logm)^3\big)$, and each sample can be generated by Algorithm 4.6 in $\mathcal{O}\big(nclogc\big)$ number of steps where $a := \max_i p_i$, $b := \max_i q_i$ and $c := \sum_i p_i = \sum_j q_j$.

4.2.2. Simple Graphs

The idea introduced in the previous section for bipartite graphs is transformed and adapted for simple graphs, and a new uniform sampling and exact counting algorithm is proposed for simple graphs with degree sequences. Recall that $\mathcal{G}^{\boldsymbol{d}}$ is the set of all simple graphs realizing the degree sequence $\boldsymbol{d} = (d_1, ..., d_n) \in \mathbb{N}^n$, and let $\mathcal{BP}$ be the set consisting of all bipartite graphs. In this section, the degree sequence $\boldsymbol{d}$ will be assumed to be non-increasing so $d_i \geq d_j$ for all $1 \leq i \leq j \leq n$.

Let $\varphi : \mathcal{G}^{\boldsymbol{d}} \to \mathcal{BP}$ be a function mapping every simple graph $G = (U; E) \in \mathcal{G}^{\boldsymbol{d}}$ with labels $\{1, ..., n\}$ to the bipartite graph $G^* = (U^+, U^-; E^*)$ where $U^+ = \{1^+, 2^+, ..., n^+\}$, $U^- = \{1^-, 2^-, ..., n^-\}$ and $E^* = \{\{i^+, j^-\}, \{i^-, j^+\} : \textit{ for all } \{i, j\} \in E\}$. Let the image of the set $\mathcal{G}^{\boldsymbol{d}}$ under the function φ be denoted by $\mathcal{SBP}^{(\boldsymbol{d},\boldsymbol{d})}$. Note that each element in the set $\mathcal{SBP}^{(\boldsymbol{d},\boldsymbol{d})}$ is a bipartite graph with partite sets $\{1^+, ..., n^+\}$ and $\{1^-, ..., n^-\}$, and each graph is a realization of the degree sequence $\{d_1^+, ..., d_n^+, d_1^-, ..., d_n^-\}$ where $d_i^+ = d_i^- = d_i$ for all $1 \leq i \leq n$. Hence, $\mathcal{SBP}^{(\boldsymbol{d},\boldsymbol{d})}$ is a proper subset of the set of bipartite graphs $\mathcal{BP}$.

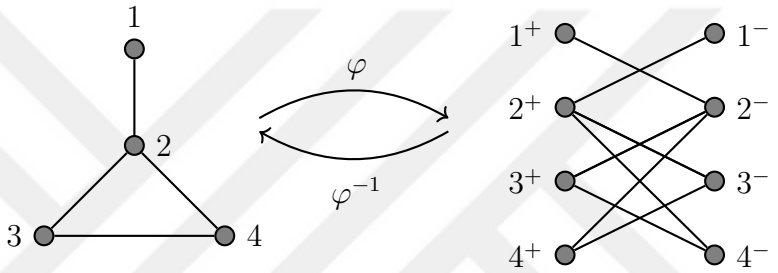


Figure 4.7. Illustration of the Function φ

Proposition 4.7. *The map $\varphi : \mathcal{G}^{\boldsymbol{d}} \to \mathcal{SBP}^{(\boldsymbol{d},\boldsymbol{d})}$ is a bijective function.*

Proof. Let $\mathcal{G}$ and $\mathcal{H}$ be isomorphic simple graphs with vertex set $\{1, ..., n\}$. Then the function $\varphi(\cdot)$ maps the graphs $\mathcal{G}$ and $\mathcal{H}$ to bipartite graphs $\widetilde{\mathcal{G}}$ and $\widetilde{\mathcal{H}}$ with partite sets $\{1^+, ..., n^+\}$ and $\{1^-, ..., n^-\}$. If the edges $\{i^+, j^-\}$ and $\{i^-, j^+\} \in E(\widetilde{\mathcal{G}})$, then $\{i, j\} \in E(\mathcal{G})$ by the definition of φ, so $\{i, j\} \in E(\mathcal{H})$ since $\mathcal{G} \cong \mathcal{H}$. Therefore, the edges $\{i^+, j^-\}$ and $\{i^-, j^+\}$ are also elements of $E(\widetilde{\mathcal{H}})$ by the definition of φ. Similarly, it can be seen that if $\{i^+, j^-\}$ and $\{i^-, j^+\} \in E(\widetilde{\mathcal{H}})$, then $\{i^+, j^-\}$ and $\{i^-, j^+\} \in E(\widetilde{\mathcal{G}})$ by the same argument. Hence, φ is a well-defined function.

It is clear that φ is a surjective function, since $\varphi[\mathcal{S}] = \mathcal{SBP}^{(\boldsymbol{d},\boldsymbol{d})}$ by initial assumption. Let $\varphi(\mathcal{G})$ and $\varphi(\mathcal{H})$ be isomorphic graphs in $\mathcal{SBP}^{(\boldsymbol{d},\boldsymbol{d})}$, and relabel (if necessary) the partite sets of the graphs $\varphi(\mathcal{G})$ and $\varphi(\mathcal{H})$ with labels $\{1^+, 2^+, ..., n^+\}$ and $\{1^-, 2^-, ..., n^-\}$ such that the edges $\{i^+, i^+\}$ and $\{i^-, i^-\}$ are not elements of $E(\varphi(\mathcal{G}))$ and $E(\varphi(\mathcal{H}))$ for any i^+ and i^-.

Then, the sets $E(\mathcal{G})$ and $E(\mathcal{H})$ consist of the edges $\{i,j\}$ for every pair of the edges $\{i^+,j^-\}$ and $\{i^-,j^+\}$ in $E(\varphi(\mathcal{G}))$ or in $E(\varphi(\mathcal{H}))$ by the definition of φ. Therefore, $\{i,j\} \in E(\mathcal{G})$ if and only if $\{i,j\} \in E(\mathcal{H})$, so φ is also an injective function. □

The cardinality of the sets $\mathcal{G}^{\boldsymbol{d}}$ and $\mathcal{SBP}^{(\boldsymbol{d},\boldsymbol{d})}$ are equal by Proposition 4.7 so the rest of this section is devoted to counting the elements in the set $\mathcal{SBP}^{(\boldsymbol{d},\boldsymbol{d})}$ rather than the simple graphs realizing the degree sequence $\boldsymbol{d} = (d_1, ..., d_n) \in \mathbb{N}^n$.

Before presenting the enumeration algorithm for simple graphs with degree sequence, some definitions and notations are needed for the description of the method. Let $\boldsymbol{p} = (p_1, ..., p_n) \in \mathbb{N}^n$ be an ordered sequence with $p_1 \geq p_2 \geq ... \geq p_n$, and the function $\omega_x(\cdot,\cdot)$ be defined by

$$\omega_x(\boldsymbol{p},k) = \max(\{i \in \mathbb{N}_{>0} : \left|\{j : p_j = x,\ and\ i \leq j \leq n\}\right| = k\} \cup \{0\})$$

for some $x \in \{p_1, ..., p_n\}$. In other words, the function gives the index of the k-th element equal to x from the end of the finite sequence $\boldsymbol{p}$, and gives 0, if the sequence $\boldsymbol{p}$ does not contain k number of elements equal to x. The function $\widetilde{\xi}_x(\cdot,\cdot)$ is defined by $\widetilde{\xi}_x(\boldsymbol{p},k) = (\widetilde{p}_1, \widetilde{p}_2, ..., \widetilde{p}_{n-1}, \widetilde{p}_n)$ where

$$\widetilde{p}_i = \begin{cases} p_i & p_i \neq x \\ p_i - 1 & p_i = x,\ and\ i \geq \omega_x(\boldsymbol{p},k) \\ p_i & p_i = x,\ and\ i < \omega_x(\boldsymbol{p},k) \end{cases}$$

Let $\boldsymbol{s} = (s_1, ..., s_a) \in \mathbb{N}^a$ be a sequence such that $\boldsymbol{s} \leq \overline{\boldsymbol{p}}$ where $a = \max_i p_i$, and let the function $\xi(\boldsymbol{p},\boldsymbol{s})$ be defined by $\widetilde{\xi}_a(\cdots(\widetilde{\xi}_2(\widetilde{\xi}_1(\boldsymbol{p},s_1),s_2),\cdots),s_a)$, which updates the last s_j elements equal to j of the sequence $\boldsymbol{p} = (p_1, ..., p_n)$ to $p_k - 1$ for each term $s_j \in \{s_1, ..., s_a\}$. Since the function $\xi(\cdot,\cdot)$ is defined with the compositions of $\widetilde{\xi}_j$'s in increasing order with respect to their indicies j, any coordinate of the sequence $\boldsymbol{p}$ can be modified by only one function $\widetilde{\xi}_j$ for some $j \in \{1, ..., a\}$.

Example 4.8. *Let* $\boldsymbol{p} = (3,2,2,1,1)$, *and* $\boldsymbol{s} = (2,1,0) \leq \overline{\boldsymbol{p}} = (2,2,1)$. *Then,* $\xi(\boldsymbol{p},\boldsymbol{s}) = (3,2,1,0,0)$, *since* $\widetilde{\xi}_1(\boldsymbol{p},s_1) = \widetilde{\xi}_1((3,2,2,1,1),2) = (3,2,2,0,0)$, *and* $\widetilde{\xi}_2((3,2,2,0,0),s_2) = \widetilde{\xi}_2((3,2,2,0,0),1) = (3,2,1,0,0)$.

By Proposition 4.7, for each labeled simple graph with the degree sequence $\boldsymbol{d} = (d_1,...,d_n)$, there is a corresponding bipartite graph in the set $\mathcal{SBP}^{(\boldsymbol{d},\boldsymbol{d})}$, and each bipartite graph in $\mathcal{SBP}^{(\boldsymbol{d},\boldsymbol{d})}$ can be represented by a symmetric binary matrix with zero-diagonal elements and margin sums equal to the degree sequence $\boldsymbol{d} = (d_1,...,d_n)$. Thus, a simple graph with a given degree sequence can be generated by constructing such a binary matrix with the same row and column sums.

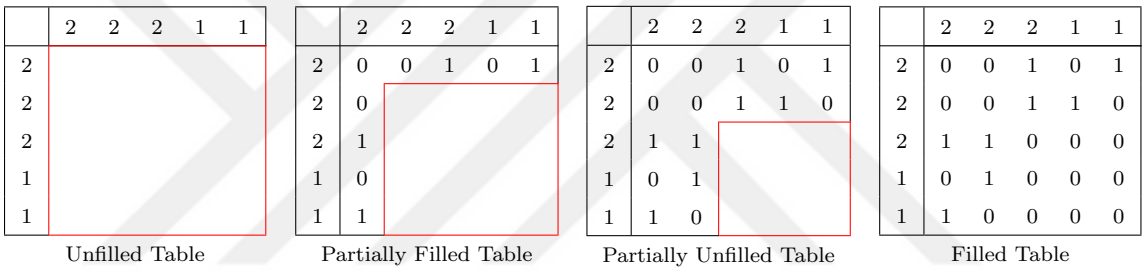


Figure 4.8. Construction of a Symmetric Binary Matrix with Zero-Diagonals

For instance, in Figure 4.8, a construction of a simple graph with the degree sequence $(2,2,2,1,1)$ is illustrated. The first row is chosen as $\boldsymbol{u} = (0,0,1,0,1)$, so the first column must be also equal to $\boldsymbol{u} = (0,0,1,0,1)$. Then, the margin sums are updated to $\mathcal{L}(\boldsymbol{d}-\boldsymbol{u}) = (2,1,1,0)$, and the same procedure can be repeated until the table is completely filled. Let $\kappa(\boldsymbol{d})$ be the number of simple graphs realizing a given degree sequence $\boldsymbol{d}$. Note that if any permutation $\boldsymbol{v}$ of a chosen row vector $\boldsymbol{u}$ satisfies the equality $\overline{\boldsymbol{d}-\boldsymbol{v}} = \overline{\boldsymbol{d}-\boldsymbol{u}}$, then $\kappa(\mathcal{L}(\boldsymbol{d}-\boldsymbol{u}) = \kappa(\mathcal{L}(\boldsymbol{d}-\boldsymbol{v}))$, in other words, the possible number of binary matrices for the unfilled part of the table is not affected by permuting the positions of 1's situated in the locations whose column sums are equal. The first entry of a chosen vector $\boldsymbol{u}$ must be 0 due to the structure of the matrix being constructed, so the column sum sequence can be considered as $\boldsymbol{r} := (0,\mathcal{L}\boldsymbol{d}) = (0,d_2,...,d_n)$ instead of the vector $\boldsymbol{d}$ at each step.

By Lemma 4.4, for a given pair of vectors $(\boldsymbol{u}, \boldsymbol{d}) \in \{0,1\}^n \times \mathbb{N}^n$ holding the inequality $\boldsymbol{u} \leq \boldsymbol{r} := (0, \mathcal{L}\boldsymbol{d})$, a sequence $\boldsymbol{s} \in \mathbb{N}^a$ satisfying $\overline{\boldsymbol{r} - \boldsymbol{u}} = \overline{\boldsymbol{r}} \backslash \boldsymbol{s}$ can be found where $a := \max_i r_i$. Therefore, the number of such permutations of a chosen vector $\boldsymbol{u}$ can be computed by $\binom{\boldsymbol{r}}{\boldsymbol{s}} = \binom{r_1}{s_1} \cdots \binom{r_a}{s_a}$ similar to the idea introduced in the previous section.

Hence, the number of all possible vectors for a row or a column can be computed by determining the different $\mathbb{N}$-valued vectors $\boldsymbol{s} \in \mathcal{C}^{\overline{\boldsymbol{r}}}(d_1)$. However, after determining a vector $\boldsymbol{s}$, the row sum sequence must be also updated since the table is symmetric. Note that the vector $\boldsymbol{v} := \boldsymbol{d} - \xi(\boldsymbol{d}, \boldsymbol{s})$ is a permutation of $\boldsymbol{u}$ satisfying $\overline{\mathcal{L}(\boldsymbol{d} - \boldsymbol{u})} = \overline{\boldsymbol{r}} \backslash \boldsymbol{s}$ since $\overline{\mathcal{L}(\boldsymbol{d} - \boldsymbol{v})} = \overline{\mathcal{L}(\boldsymbol{d} - (\boldsymbol{d} - \xi(\boldsymbol{d}, \boldsymbol{s})))} = \overline{\mathcal{L}\xi(\boldsymbol{d}, \boldsymbol{s})} = \overline{\xi(\mathcal{L}\boldsymbol{d}, \boldsymbol{s})} = \overline{\boldsymbol{r} - \boldsymbol{u}} = \overline{\mathcal{L}(\boldsymbol{d} - \boldsymbol{u})}$, so $\overline{\xi(\mathcal{L}\boldsymbol{d}, \boldsymbol{s})} = \overline{\boldsymbol{r}} \backslash \boldsymbol{s}$, and $\kappa(\xi(\mathcal{L}\boldsymbol{d}, \boldsymbol{s})) = \kappa(\mathcal{L}(\boldsymbol{d} - \boldsymbol{u}))$. Therefore, the row sums sequence $\boldsymbol{d}$ can be updated to $\xi(\mathcal{L}\boldsymbol{d}, \boldsymbol{s})$ after each step for a chosen vector $\boldsymbol{s} \in \mathcal{C}^{\overline{\boldsymbol{r}}}(d_1)$ in order to not violate the non-increasing order of the row sums.

Now the recursive formula to compute the number of realizations of a given degree sequence can be presented.

Theorem 4.9. *The number of simple graphs with a given non-increasing degree sequence $\boldsymbol{d} = (d_1, ..., d_n) \in \mathbb{N}^n$, is given by $\mathcal{K}(\boldsymbol{d}, \overline{\boldsymbol{d}}) := \kappa(\boldsymbol{d})$, and the function $\mathcal{K}(\cdot, \cdot)$ is defined by*

$$\mathcal{K}(\boldsymbol{d}, \overline{\boldsymbol{d}}) = \sum_{\boldsymbol{s} \in C^{\overline{r}}(d_1)} \binom{\overline{\boldsymbol{r}}}{\boldsymbol{s}} \mathcal{K}(\xi(\mathcal{L}\boldsymbol{d}, \boldsymbol{s}), \overline{\boldsymbol{r}} \backslash \boldsymbol{s})$$

where the sequence $\boldsymbol{r} := (0, \mathcal{L}\boldsymbol{d}) = (0, d_2, ..., d_n)$.

Proof. Let $\boldsymbol{d} = (d_1, ..., d_n) \in \mathbb{N}^n$, then

$$\begin{aligned}
\mathcal{K}(\boldsymbol{d}, \overline{\boldsymbol{d}}) = \kappa(\boldsymbol{d}) &\overset{1}{=} \sum_{\boldsymbol{u} \in C^{\boldsymbol{r}\wedge 1}(d_1)} \kappa(\mathcal{L}(\boldsymbol{d} - \boldsymbol{u})) \\
&\overset{2}{=} \sum_{\boldsymbol{s} \in C^{\overline{\boldsymbol{r}}}(d_1)} \sum_{\substack{\boldsymbol{u} \in C^{\boldsymbol{r}\wedge 1}(d_1) \\ \overline{r-u} = \overline{r} \setminus s}} \kappa(\mathcal{L}(\boldsymbol{d} - \boldsymbol{u})) \\
&\overset{3}{=} \sum_{\boldsymbol{s} \in C^{\overline{\boldsymbol{r}}}(d_1)} \sum_{\substack{\boldsymbol{u} \in C^{\boldsymbol{r}\wedge 1}(d_1) \\ \overline{r-u} = \overline{r} \setminus s}} \mathcal{K}(\mathcal{L}(\boldsymbol{d} - \boldsymbol{u}), \overline{\mathcal{L}(\boldsymbol{d} - \boldsymbol{u})}) \\
&\overset{4}{=} \sum_{\boldsymbol{s} \in C^{\overline{\boldsymbol{r}}}(d_1)} \binom{\overline{\boldsymbol{r}}}{\boldsymbol{s}} \mathcal{K}(\xi(\mathcal{L}\boldsymbol{d}, \boldsymbol{s}), \overline{\boldsymbol{r}} \setminus \boldsymbol{s})
\end{aligned}$$

In the first equality, d_1 possible places other than the first cell are chosen to construct the first row or column of a binary matrix. In the second line, as it is applied in the proof of Theorem 4.6, the sum in the previous equality is divided into disjoint subsets based on the fact that for any $\boldsymbol{u} \in C^{r\wedge 1}(d_1)$ there exists $\boldsymbol{s} \in C^{\overline{\boldsymbol{r}}}(d_1)$ satisfying $\overline{\boldsymbol{r} - \boldsymbol{u}} = \overline{\boldsymbol{r}} \setminus \boldsymbol{s}$ by Lemma 4.4. The third line follows from the definition of the function $\mathcal{K}(\cdot, \cdot)$. Note that $\mathcal{L}\boldsymbol{d} = \mathcal{L}\boldsymbol{r}$, and $\overline{\mathcal{L}(\boldsymbol{r} - \boldsymbol{u})} = \overline{\boldsymbol{r} - \boldsymbol{u}}$ since the first elements of the sequences $\boldsymbol{r}$ and $\boldsymbol{u}$ are zero so $\mathcal{K}(\mathcal{L}(\boldsymbol{d} - \boldsymbol{u}), \overline{\mathcal{L}(\boldsymbol{d} - \boldsymbol{u})}) = \mathcal{K}(\mathcal{L}(\boldsymbol{d} - \boldsymbol{u}), \overline{\mathcal{L}(\boldsymbol{r} - \boldsymbol{u})}) = \mathcal{K}(\mathcal{L}(\boldsymbol{d} - \boldsymbol{u}), (\overline{\boldsymbol{r} - \boldsymbol{u}}))$. The last line follows from Lemma 4.5, and from the fact that the expression $\kappa(\mathcal{L}(\boldsymbol{d} - \boldsymbol{u})) = \kappa(\mathcal{L}(\boldsymbol{d} - \boldsymbol{v}))$ holds if the permutation $\boldsymbol{v}$ of the vector $\boldsymbol{u}$ satisfies the equality $\overline{\mathcal{L}(\boldsymbol{d} - \boldsymbol{u})} = \overline{\mathcal{L}(\boldsymbol{d} - \boldsymbol{v})}$. The vector $\boldsymbol{v}$ defined by $\boldsymbol{d} - \xi(\boldsymbol{d}, \boldsymbol{s})$ is such a permutation of the vector $\boldsymbol{u}$, because $\overline{\mathcal{L}(\boldsymbol{d} - \boldsymbol{v})} = \overline{\mathcal{L}(\boldsymbol{d} - (\boldsymbol{d} - \xi(\boldsymbol{d}, \boldsymbol{s})))} = \overline{\mathcal{L}\xi(\boldsymbol{d}, \boldsymbol{s})} = \overline{\xi(\mathcal{L}\boldsymbol{d}, \boldsymbol{s})} = \overline{\mathcal{L}(\boldsymbol{d} - \boldsymbol{u})}$. Hence, $\mathcal{K}(\mathcal{L}(\boldsymbol{d} - \boldsymbol{u}), (\overline{\boldsymbol{r} - \boldsymbol{u}})) = \mathcal{K}(\xi(\mathcal{L}\boldsymbol{d}, \boldsymbol{s}), \overline{\boldsymbol{r} - \boldsymbol{u}}) = \mathcal{K}(\xi(\mathcal{L}\boldsymbol{d}, \boldsymbol{s}), \overline{\boldsymbol{r}} \setminus \boldsymbol{s})$ by Lemma 4.4. □

The following algorithm is based on the idea of Theorem 4.9, and computes the exact number of realizations of a given degree sequence.

Input: A graphical degree sequence $\boldsymbol{d} = (d_1, ..., d_n) \in \mathbb{N}^n$

Output: The number of simple graphs realizing the given sequence $\boldsymbol{d}$

Storage: Lookup table storing intermediate results, initialized with $\mathcal{K}(0, 0) = 1$

1: Check the lookup table, if $\mathcal{K}(\boldsymbol{d}, \overline{\boldsymbol{d}})$ was cached before, return it.

2: If there exists no simple graph realizing the input sequence, cache the result, and return 0

3: For each possible sequence $\boldsymbol{s} \in \mathcal{C}^{\overline{r}}(d_1)$, compute the value of $\mathcal{K}(\xi(\mathcal{L}\boldsymbol{d}, \boldsymbol{s}), \overline{\boldsymbol{r}} \backslash \boldsymbol{s})$ where $\boldsymbol{r} = (0, \mathcal{L}\boldsymbol{d})$

4: Evaluate the sum $\sum_{\boldsymbol{s} \in C^{\overline{r}}(d_1)} \binom{\overline{r}}{\boldsymbol{s}} \mathcal{K}(\xi(\mathcal{L}\boldsymbol{d}, \boldsymbol{s}), \overline{\boldsymbol{r}} \backslash \boldsymbol{s})$, and cache the result.

5: Return the result found in the previous step (4).

Figure 4.9. Exact Enumeration of Simple Graphs with Prescribed Degree Sequences

As it is described for the algorithm in Figure 4.4, the third and fourth steps can be combined, and only the sequences satisfying slightly modified Gale-Ryser conditions can be considered. Similarly, Algorithm 4.9 traverses the nodes of a tree by starting from the root node $(\boldsymbol{d}, \bar{\boldsymbol{d}})$. Let $\mathcal{S}(\boldsymbol{d}, \bar{\boldsymbol{d}})$ be the set of all pairs $(\boldsymbol{\eta}, \bar{\boldsymbol{\eta}})$ traversed by Algorithm 4.9 where $(\boldsymbol{d}, \bar{\boldsymbol{d}})$ is the root node. Note that a pair $(\boldsymbol{\eta}, \overline{\boldsymbol{\eta}})$ is a child of $(\boldsymbol{h}, \overline{\boldsymbol{h}})$ if $\boldsymbol{\eta} = \xi(\mathcal{L}\boldsymbol{h}, \boldsymbol{s})$ for some $\boldsymbol{s} \in \mathcal{C}^{\overline{\mathcal{L}\boldsymbol{h}}}(h_1)$. The following sampling algorithm allows one to uniformly draw samples from the set of all realizations of a given degree sequence.

Input: A graphical degree sequence $\boldsymbol{d} = (d_1, ..., d_n)$

Output: A uniformly drawn sample from the set of all zero-diagonal binary matrices satisfying the margin $(\boldsymbol{d}, \boldsymbol{d})$

Storage: Lookup table storing the results of $\mathcal{K}(\boldsymbol{\eta}, \overline{\boldsymbol{\eta}})$ for each pair $(\boldsymbol{\eta}, \overline{\boldsymbol{\eta}}) \in \mathcal{S}(\boldsymbol{d}, \overline{\boldsymbol{d}})$

1: Initialize $(\boldsymbol{\eta}, \overline{\boldsymbol{\eta}})$ with $(\boldsymbol{d}, \overline{\boldsymbol{d}})$

2: If $(\boldsymbol{\eta}, \overline{\boldsymbol{\eta}}) = (\boldsymbol{0}, \boldsymbol{0})$, terminate with the set of constructed rows.

3: Select a child $(\xi(\mathcal{L}\boldsymbol{\eta}, \boldsymbol{s}), \overline{\boldsymbol{r}} \backslash \boldsymbol{s}) \in \mathcal{S}(\boldsymbol{d}, \overline{\boldsymbol{d}})$ of $(\boldsymbol{\eta}, \overline{\boldsymbol{\eta}})$ with a probability proportional to its count times the number of vectors $\boldsymbol{u} \in \mathcal{C}^{\boldsymbol{r} \wedge 1}(\eta_1)$ such that $\overline{\boldsymbol{r} - \boldsymbol{u}} = \overline{\boldsymbol{r}} \backslash \boldsymbol{s}$, where $\boldsymbol{r} = (0, \mathcal{L}\boldsymbol{\eta})$.

4: Uniformly choose a row $\boldsymbol{u} \in \mathcal{C}^{\boldsymbol{r} \wedge 1}(\eta_1)$ from the set $\{\boldsymbol{u} : \overline{\boldsymbol{r} - \boldsymbol{u}} = \overline{\boldsymbol{r}} \backslash \boldsymbol{s}\}$ at random.

5: Set $(\boldsymbol{\eta}, \overline{\boldsymbol{\eta}})$ to $(\xi(\mathcal{L}\boldsymbol{\eta}, \boldsymbol{s}), \overline{\boldsymbol{r}} \backslash \boldsymbol{s})$.

6: Choose an index position of the vector $\boldsymbol{u}$

7: Return to the step (2).

Figure 4.10. Uniform Sampling of Simple Graphs with a Given Degree Sequence

Notice that the uniform sampling Algorithm 4.10 returns a symmetric binary matrix with zero-diagonals as an output for a given graphical degree sequence, and an output matrix $M = (m_{ij})$ can be easily converted into a simple graph by placing edges between the vertices i and j for each entry m_{ij} or m_{ji} equal to 1.

5. APPLICATIONS AND SIMULATIONS

In this chapter, the algorithms presented in the previous sections are tested in several examples to illustrate their applications in different fields.

5.1. Enumeration of Simple and Bipartite Graphs with Given Degree Sequences

For a given degree sequence $\boldsymbol{d} = (d_1, ..., d_n)$, the natural question is that how many simple graphs are there realizing the degree sequence $\boldsymbol{d} = (d_1, ..., d_n)$. However, there is no known simple formula so far to count the number of graphs in $\mathcal{G}^{\boldsymbol{d}}$.

In 1978, Bender and Canfield [38] gave the first asymptotic formula for the case where $\max_i d_i$ is bounded. Later, the constraints on the degrees were improved, and better asymptotic formulas were suggested [39, 40]. For instance, in 1991, McKay and Brendan [41] showed that if $d_{max} = o(M_1^{1/3})$, then

$$|\mathcal{G}^{\boldsymbol{d}}| \sim \frac{M_1!}{(\frac{M_1}{2})!2^{\frac{M_1}{2}} d_1! \cdots d_n!} exp\left(\frac{M_3^2}{6M_1^3} + \frac{M_2^4}{4M_1^5} - \frac{M_2^2 M_3}{2M_1^4} - \frac{M_2^2}{4M_1^2} - \frac{M_2}{2M_1} + O\left(\frac{d_{max}^3}{M_1}\right)\right)$$

as $M_1 \to \infty$, where $d_{max} = \max_i d_i$, $[d]_j = d(d-1)\cdots(d-j+1)$, and $M_j = \sum_i [d_i]_j$.

As it is described in the previous Section 4.2.2, Theorem 4.9 can be used to exactly compute the number of simple graphs in the set $\mathcal{G}^{\boldsymbol{d}}$. In the tables below, the estimated results found by the importance sampling technique in Subsection 4.1.1 and the exact values computed by the proposed method are given for some regular and arbitrary degree sequences in order to compare the two different algorithms.

Table 5.1. Enumeration of Regular Graphs

n	k-regular	Importance Sampling	Proposed Method
4	3-regular	1.0 ± 0.0	1
6	3-regular	70,625 ± 0.895	70
8	4-regular	19.133,298 ± 325,706	19.355
10	4-regular	(6,6373 ± 0,1149) $\times 10^7$	66.462.606
12	9-regular	(3.5896 ± 0,1248) $\times 10^7$	34.944.085
14	9-regular	(6,4632 ± 0,2664) $\times 10^{15}$	6.551.246.596.501.035
16	11-regular	(1,4806 ± 0,0765) $\times 10^{20}$	155.243.722.248.524.067.795
18	11-regular	(6,0905 ± 0,3299) $\times 10^{30}$	5.997.229.769.947.050.271.535.917.422.040

In Table 5.1, the results for some regular graphs are listed, and Table 5.2 contains the number of graphs satisfying the degree sequence indicated in its first column. As it is shown in the tables, the estimated results are very close to their exact values, and only 1000 samples are used in finding the approximate values with importance sampling algorithm.

Table 5.2. Enumeration of Graphs with Arbitrary Degree Sequences

Degree Sequence	Importance Sampling	Proposed Method
(4,2,5,2,2,3)	3.0039 ± 0.0095	3
(2,4,2,4,5,4,1)	11.9892 ± 0.0570	12
(2,2,3,1,7,2,5,3,5)	212.5956 ± 5.1241	215
(4,6,6,5,2,1,3,8,4,1,4)	$(1,17523 \pm 0,02909) \times 10^5$	117.697
(1,4,1,5,9,2,6,5,3,5,8,9)	$(9,386 \pm 0,726) \times 10^3$	9.276
(5,3,3,4,4,2,2,3,6)	$(29,959 \pm 0,342) \times 10^3$	29.705
(7,5,4,8,4,3,5,2,4,4)	$(1,38785 \pm 0,03269) \times 10^5$	139.370

5.2. Hypothesis Testing

In ecology, researchers use *co-occurence tables* to analyze the patterns of species in different ecological locations, which are often used to indicate the presence or absence of species on certain ecological locations. For instance, Table 5.3 shows the distribution of 17 reptiles species in Tasmania and the Bass Strait area [42] where $1's$ represent the occurrence of species on the corresponding islands. Only the part of the table containing islands and groups of islands were considered and the remaining part was excluded. Columns and rows of the table correspond to reptile species and names of locations which they live on, respectively.

Table 5.3. Distribution of Island Reptile Species In Southern Australia

	A. diemensis	Leiolopisma delicata	L. entrecasteauxii	L. metallica	L. ocellata	L. pretiosa	L. trilineata	L. sp. nov.	Lerista bougainwillii	Pseudemoia sp. nov.	S. tympanum	Egernia whitei	Tiliqua casuarinae	T. nigrolutea	A. superbus	Drysdalia coronoides	Notechis ater
Pedra Branca Island	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0
Maatsuyker Island	0	0	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0
Tasmania	1	1	1	1	1	1	1	1	1	0	0	1	1	1	1	1	1
Albatros Island	0	0	1	1	0	1	1	0	0	0	0	1	0	1	1	1	1
Furneaux Group	0	0	0	1	0	1	0	0	0	0	0	0	0	0	0	0	0
Curis Island	1	0	1	1	1	1	1	0	1	0	0	1	0	1	1	1	1
Kent Group	0	0	0	1	0	0	0	0	1	0	0	1	0	0	0	1	0
Hogan Group	0	0	0	1	0	0	1	0	1	0	0	1	0	1	0	1	0
Rodondo Island	0	0	1	1	0	0	1	0	0	0	0	1	0	1	0	0	0
Brisbane-Adelaide	0	0	0	1	0	0	0	0	0	0	1	1	0	0	0	0	0

Species usually share their ecological areas with other species, and they are positively or negatively interact with them.

For instance, they may compete for the same food resource in a certain area, or co-operate to benefit from each others behaviors like mutualistic relationships. These interactions among different species may have caused the emergence of a particular pattern in an ecological environment, but this distribution may have been also occurred completely by chance. In other words, the pattern of species could not be much different than what would be expected if there was no interactions among them. Robert and Stone [43] proposed a simple test statistic which is given by

$$\bar{S}^2 = \frac{1}{n(n-1)} \sum_{i \neq j} s_{ij}^2$$

where n is the number of species, and $S = (s_{ij}) = M^T M$, and M is a $m \times n$ occurrence matrix. Intuitively, the statistic measures the interaction between the distribution of different species over a set of certain ecological locations. The lower value of $\bar{S}^2$ implies that the pattern of a species has not much influence on the distribution of the other species. Similar to the study of Chen *et al.* [2], the statistic can be used to test the null hypothesis that the reptile species are distributed randomly on the islands. Note that the observed statistic of the co-occurrence table is 5.4485. To carry out the hypothesis test, 100.000 samples satisfying the margin sums of the table are generated from the uniform distribution, and the histogram of the test statistic is depicted in Figure 5.1.

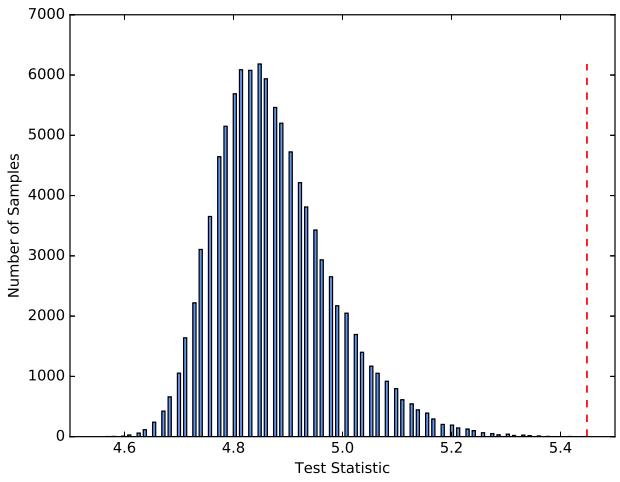


Figure 5.1. Histogram of the Test Statistic $\overline{S^2}$ for 100.000 Samples

The red dashed line in Figure 5.1 indicates the value of $\bar{S}^2$ for the reptile species table. The SIS and exact sampling algorithms described in Subsections 4.1.2 and 4.2.1 estimated the p-value to be 0.09×10^{-3}, and 0.02×10^{-3}, respectively. Hence, the results demonstrate that the observed statistic is not consistent with the null hypothesis and there is enough evidence to reject the null hypothesis in favor of the alternative hypothesis.

5.3. Network Analysis

Graph measures are important elements in the network analysis, which are generally used to characterize the structural properties of a graph. Any feature of an observed graph can be compared with the features of an ensemble of graphs sharing the same degree sequence so that deviation of the observed graph's feature from the mean can be determined. Consider the friendship network in Figure 1.1 which consists of 23 vertices of degrees ranging from 2 to 14, and note that the degree sequence of the graph does not admit the power-law distribution.

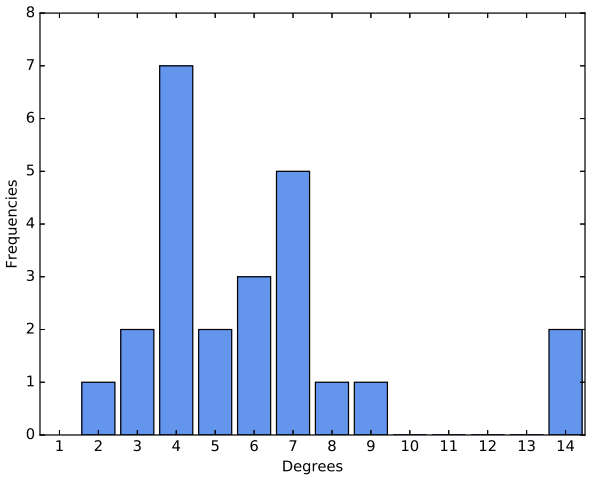


Figure 5.2. Degree Frequencies of the Friendship Network of Freshman Students

Clustering coefficient is one of the widely used metrics, which intuitively measures how connected are the neighbors of vertices in the graph.

The metric has different variants, and *local clustering coefficient* is a version proposed by Watts and Strogatz in 1998 [10]. It is the ratio between the number of edges among the neighbors of a vertex, and the largest possible number of edges among them. The formal definition of local clustering coefficient of a vertex v is given by

$$cc(v) := \frac{|\{\{u, w\} \in E(G) : u, w \in \mathcal{N}(v)\}|}{\binom{k}{2}}$$

where $k = |\mathcal{N}(u)|$, and the clustering coefficient of the whole graph is the mean of local clustering coefficients of vertices of the graph.

$$\overline{cc}(G) = \frac{1}{N} \sum_{v \in V(G)} cc(v)$$

Figure 5.3 below shows the histogram of local clustering coefficients of graphs sharing the same degree sequence with the friendship network, and the red dashed line indicates the clustering coefficient of the friendship network, which is equal to $0,6046$. The friendship network exhibits high clustering coefficient with respect to the other generated graphs.

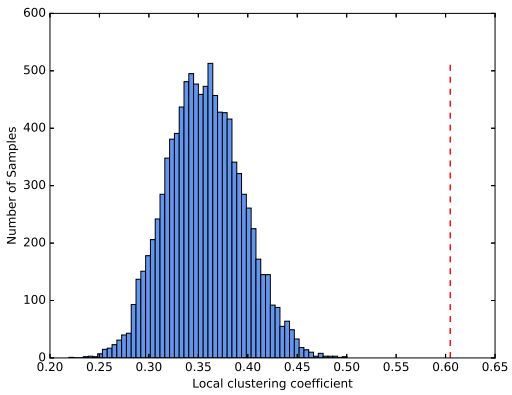


Figure 5.3. Clustering Coefficients

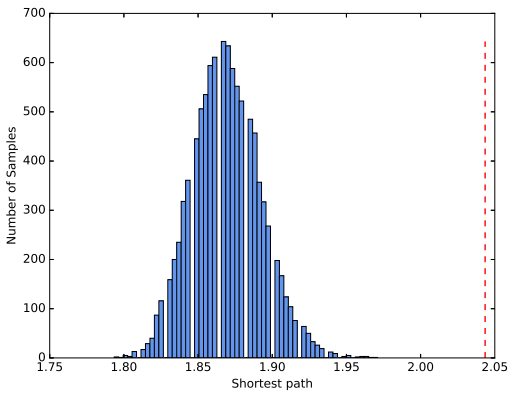


Figure 5.4. Avg. Shortest Path Lengths

The average shortest path length is another important characteristic of networks which is defined by

$$asp(G) = \sum_{u,v \in V(G)} \frac{d(u,v)}{n(n-1)}$$

where $n := |V(G)|$, and $d(u,v)$ is the shortest path length between the vertices u and v. A vertex $u \in V(G)$ is *reachable* from another vertex $v \in V(G)$, if there exists a path between u and v, and if the vertex u is not reachable from v, $d(u,v) = 0$. Unlike the small-world networks exhibiting small average shortest path length along with high clustering coefficient, the average shortest path length of the friendship network is 2.0435, and it is higher than the expected value of the average shortest path of graphs with the same degree sequence.

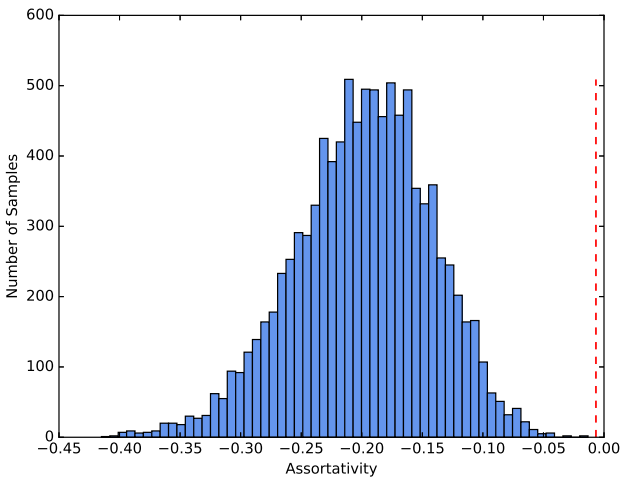


Figure 5.5. Assortativity Coefficients

Assortativity coefficient quantifies the tendency of vertices to be adjacent to other vertices of similar degrees, which is simply the Pearson correlation coefficient of degrees of vertices connected to each other. In 2002, Newman [44] introduced the assortativity coefficient, and showed that many social networks admit assortative mixing, which means that there is a correlation between vertices of similar degree.

The histogram in Figure 5.5 points out that the random graphs generated by the same degree sequence shows mostly a disassortative pattern like technological and biological networks, but the assortativity coefficient of the friendship network is -0.006651, and it almost shows no assortative mixing pattern.

The method developed in Section 4.2.2 are used to obtain uniformly distributed simple graphs, and 10.000 samples are generated for each experiment. Similarly, other graph measures can be used to interpret the friendship network, but it can be roughly said that the observed network shows a very different characteristic than the graphs with the same degree sequence.

6. CONCLUSION

In this thesis, various sampling approaches for simple and bipartite graphs were investigated, and they were mainly classified into two parts. The first part was devoted to Markov chain Monte Carlo methods, and a well-known edge switching based Markov chain method was introduced for simple graphs. It was proved that the chain converges to the uniform distribution over all realizations of a given degree sequence, and the chain was used for obtaining uniformly distributed samples.

In the second part, direct construction methods were considered for simple and bipartite graphs. Since directly generating uniformly distributed samples is a complicated task, various Monte Carlo methods were applied to the problem and were used to estimate the properties of the uniform distribution. The importance sampling algorithm proposed by Blitzstein and Diaconis [1] and sequential importance sampling technique suggested by Chen *et al.* [2] were investigated for simple and bipartite graphs. A new uniform sampling and exact counting algorithm was developed for labeled simple graphs based on the work of Miller and Harrison [3].

The applications of the presented algorithms were illustrated in several examples. The proposed method in Subsection 4.2.2 was used to exactly count the number of graphs satisfying arbitrary graphical degree sequences. A hypothesis test was carried out for the ecological network in Table 5.3 formed by island reptile species in Australia, and the null hypothesis that the reptile species are distributed randomly on the islands was rejected. The friendship network of freshman students in Figure 1.1 was analyzed, and it was shown that the network admitted a very different characteristic than the most of the graphs sharing the same degree sequence with the network.

The main disadvantage of the edge switching Markov chain is its mixing time, and it is still an open question waiting to be answered. The exact counting and sampling methods described in Subsections 4.2.1 and 4.2.2 mostly run slower than the approximate methods.

However, the main strength of these algorithms stems from their exactness, and the approximate methods require a large number of samples for better estimations. In the future, faster algorithms may be designed for the generation of uniformly distributed simple and bipartite graphs realizing the given degree sequences.

REFERENCES

1. Joseph Blitzstein, P. D., "A sequential importance sampling algorithm for generating random graphs with prescribed degrees", *Internet Mathematics*, Vol. 6, pp. 489–522, 2011.

2. Chen, Y., P. Diaconis, S. P. Holmes and J. S. Liu, "Sequential Monte Carlo Methods for Statistical Analysis of Tables", *Journal of the American Statistical Association*, Vol. 100, No. 469, pp. 109–120, 2005.

3. Miller, J. W. and M. T. Harrison, "Exact sampling and counting for fixed-margin matrices", *The Annals of Statistics*, Vol. 41, pp. 1569–1592, 2013.

4. Newman, M. E. J., "The Structure and Function of Complex Networks", *SIAM Review*, Vol. 45, pp. 167–256, January 2003.

5. van de Bunt, G., *Friends by choice: An actor-oriented statistical network model for friendship networks through time*, Ph.D. Thesis, 1999, naam instelling promotie: ICS Rijksuniversiteit Groningen Naam instelling onderzoek: ICS Rijksuniversiteit Groningen.

6. Van De Bunt, G. G., M. A. Van Duijn and T. A. Snijders, "Friendship Networks Through Time: An Actor-Oriented Dynamic Statistical Network Model", *Computational and Mathematical Organization Theory*, Vol. 5, No. 2, pp. 167–192, 1999.

7. A., E. R., "On Random Graphs. I", *Publicationes Mathematicae.*, Vol. 6, p. 290–297, 1959.

8. Gilbert, E. N., "Random Graphs", *The Annals of Mathematical Statistics*, Vol. 30, No. 4, pp. 1141–1144, December 1959.

9. Solomonoff, R. and A. Rapoport, "Connectivity of random nets", *The bulletin of*

mathematical biophysics, Vol. 13, No. 2, pp. 107–117, 1951.

10. Duncan J. Watts, S. H. S., "Collective dynamics of small-world networks", *Nature*, Vol. 393, pp. 440–442, 1998.

11. Barabási, A.-L. and R. Albert, "Emergence of Scaling in Random Networks", *Science*, Vol. 286, No. 5439, pp. 509–512, 1999.

12. Ravasz, E. and A.-L. Barabási, "Hierarchical organization in complex networks", *Physical Review E*, Vol. 67, p. 026112, February 2003.

13. Geng, X. and Q. Li, "Random models of scale-free networks", *Physica A: Statistical Mechanics and its Applications*, Vol. 356, No. 2–4, pp. 554 – 562, 2005.

14. D., K., "Cognitive social structures", *Social Networks*, Vol. 9, pp. 109–134, 1987.

15. Jr., J. M. R., "Simple methods for simulating sociomatrices with given marginal totals", *Social Networks*, Vol. 22, No. 3, pp. 273 – 283, 2000.

16. Newman, M. E. J., "Spread of epidemic disease on networks", *Physical Review E*, Vol. 66, p. 016128, July 2002.

17. Balakrishnan, R. and K. Ranganathan, *A Textbook of Graph Theory*, Universitext, Springer New York, 2012.

18. Diestel, R., *Graph Theory*, Springer-Verlag Berlin Heidelberg, 2010.

19. Wallis, W. D., *A Beginner's Guide to Graph Theory*, Springer-Verlag Berlin Heidelberg, 2007.

20. Bollobas, B., *Modern Graph Theory*, Springer-Verlag New York, 1998.

21. Çinar, E., *Probability and Stochastics*, Springer New York, 2011.

22. Athreya, K. B. and S. N. Lahiri, *Measure Theory and Probability Theory*, Springer New York, 2006.

23. Skorokhod, A., *Basic Principles and Applications of Probability Theory*, Springer Berlin Heidelberg, 2005.

24. Serfozo, R., *Basics of Applied Stochastic Processes*, pp. 1–83, Springer-Verlag Berlin Heidelberg, 2009.

25. Taylor, R., *Contrained switchings in graphs*, pp. 314–336, Springer Berlin Heidelberg, Berlin, Heidelberg, 1981.

26. Kannan, R., P. Tetali and S. Vempala, "Simple Markov-chain Algorithms for Generating Bipartite Graphs and Tournaments", *Random Structures and Algorithms*, Vol. 14, No. 4, pp. 293–308, July 1999.

27. Cooper, C., M. Dyer and C. Greenhill, "Sampling Regular Graphs and a Peer-to-peer Network", pp. 980–988, 2005.

28. Greenhill, C., "A polynomial bound on the mixing time of a Markov chain for sampling regular directed graphs", Vol. 18, No. 1, 2011.

29. Greenhill, C. S. and M. Sfragara, "The switch Markov chain for sampling irregular graphs and digraphs", *ArXiv e-prints*, Vol. abs/1701.07101, January 2017.

30. Miklós, E. and Soukup, "Towards random uniform sampling of bipartite graphs with given degree sequence", Vol. 20, No. 1, 2013.

31. Greenhill, C., "The Switch Markov Chain for Sampling Irregular Graphs: Extended Abstract", pp. 1564–1572, 2015.

32. Péter L. Erdös, I. M. and Z. Toroczkai, "A Decomposition Based Proof for Fast Mixing of a Markov Chain over Balanced Realizations of a Joint Degree Matrix", , 2015.

33. Erdős, P. L., I. Miklós and Z. Toroczkai, "New classes of degree sequences with fast mixing swap Markov chain sampling", *ArXiv e-prints*, January 2016.

34. Tripathi, A., S. Venugopalan and D. B. West, "A short constructive proof of the Erdős–Gallai characterization of graphic lists", *Discrete Mathematics*, Vol. 310, pp. 397–405, 2010.

35. Chen, S. X. and J. S. Liu, "Statistical Applications of the Poisson-Binomial and Conditional Bernoulli Distributions", *Statistica Sinica*, Vol. 7, No. 4, pp. 875–892, 1997.

36. Gail, M. H., J. H. Lubin and L. V. Rubinstein, "Likelihood Calculations for Matched Case-Control Studies and Survival Studies with Tied Death Times", *Biometrika*, Vol. 68, No. 3, pp. 703–707, 1981.

37. Krause, M., "A Simple Proof of the Gale-Ryser Theorem", *The American Mathematical Monthly*, Vol. 103, pp. 335–337, 1996.

38. "The asymptotic number of labeled graphs with given degree sequences", *Journal of Combinatorial Theory, Series A*, Vol. 24, No. 3, pp. 296 – 307, 1978.

39. Bollobás, B., "A Probabilistic Proof of an Asymptotic Formula for the Number of Labelled Regular Graphs", *European Journal of Combinatorics*, Vol. 1, No. 4, pp. 311 – 316, 1980.

40. McKay, B. D., "Asymptotics for symmetric 0-1 matrices with prescribed row sums", *Ars Combin*, Vol. 19, pp. 15–25, 1985.

41. McKay, B. D. and N. C. Wormald, "Asymptotic enumeration by degree sequence of graphs with degrees $o(n^{1/2})$", *Combinatorica*, Vol. 11, No. 4, pp. 369–382, 1991.

42. Rawlinson, P. A., *Biogeography and Ecology of the Reptiles of Tasmania and the Bass Strait Area*, pp. 291–338, Springer Netherlands, 1974.

43. Roberts, A. and L. Stone, "Island-sharing by archipelago species", *Oecologia*, Vol. 83, No. 4, pp. 560–567, 1990.

44. Newman, M. E., "Assortative Mixing in Networks", *Physical Review Letters*, Vol. 89, No. 20, p. 208701, October 2002.