Editors' introduction

The theory of random graphs was established during the 1950s through the pioneering work of Gilbert and subsequently of Erdős and Rényi who set its foundations. Since then, the theory has been developed vastly and is by now a central area of combinatorics. Numerous, often unexpected, ramifications have emerged, which link it to diverse areas of mathematics such as number theory, combinatorial optimization and probability theory. Since its beginning, the study of geometric and topological aspects of random graphs has become the meeting point between combinatorics and areas of probability theory, such as percolation theory and stochastic processes. Nowadays, this interface has been consolidated through numerous deep results. This has led to applications in other scientific disciplines including telecommunications, astronomy, statistical physics, biology and computer science, as well as much more recent developments such as the study of social and biological networks.

The present book is the outcome of a short course that took place at the School of Mathematics of the University of Birmingham in August 2013 and was supported by the London Mathematical Society and the Engineering and Physical Sciences Research Council. Its aim was to provide a concise overview of recent trends in the theory of random graphs, ranging from classical structural problems to geometric and topological aspects, and to introduce the participants to new powerful complex–analytic techniques and stochastic models that have led to recent breakthroughs in the field.

The theory of random graphs is nowadays part and parcel of the education of any young researcher entering the fascinating world of combinatorics. However, due to their interdisciplinary nature, the geometric and structural aspects of the theory often remain an obscure part of the education of young researchers. Moreover, the theory itself, even in its most basic forms, is often considered quite advanced to be part of undergraduate curricula, and those interested, usually learn it mostly through self-study, covering a lot of its fundamentals but not much of the more recent developments. The present book provides a self-contained and concise introduction to recent developments and techniques for classical problems in the theory of random graphs. Moreover, it covers geometric and topological aspects of the theory of random graphs and introduces the reader to the diversity and depth of the methods that have been invented in this context. Emphasis is given to powerful complex–analytic approaches as well as to sophisticated tools based on stochastic processes. To the best of our knowledge, it is the first time that such diverse approaches are put together in a single volume.

This book consists of four chapters that complement each other. These chapters are aimed to be mostly self-contained so as to be particularly accessible to researchers with limited background in the theory of random graphs. They have been developed by Prof. M. Krivelevich, Prof. K. Panagiotou, Prof. M. D. Penrose and Prof. C. McDiarmid, respectively. Each of these four academics is an internationally acclaimed expert in the field covered in their chapter.

The first chapter covers some recent developments regarding the classical problem of finding long paths and cycles in random graphs. Paths and cycles have always been between the most central and well-studied notions in graph theory. It is thus only natural they have become one of the foci of attention of the theory of random graphs, resulting in many beautiful results and inspiring methods whose importance reaches far beyond probabilistic combinatorics. This chapter reviews several basic results and approaches in long cycles and Hamiltonicity in random graphs, covering both classical theorems and recent developments.

The second chapter covers properties of random objects from graph classes with structural constraints, such as planar graphs. The corresponding random graph models are usually much harder to work with than the classical binomial random graph model. The standard methodology that is applied in this context is the use of generating functions and analytic techniques. In this chapter, a new approach based on random sampling techniques that is both technically significantly simpler and more widely applicable is presented. The chapter starts with a general presentation of the method with an application to random trees. It then proceeds with the rigorous development of the theory of combinatorial species so that general classes can be decomposed and sampled. This part contains a basic development of the so-called Boltzmann sampling, as it has been developed by Flajolet and co-authors. This method is applied to determine the typical block structure of graphs from certain classes as well as their degree sequence.

The third chapter focuses on the theory of finite graphs with nodes placed randomly in a Euclidean space and edges added to connect those pairs of nodes that are close to one another. As an alternative to classical random graph models, these geometric graphs are relevant to the modeling of real-world networks having spatial content, arising in applications in areas such as wireless communications, classification, epidemiology and astronomy. The following topics are covered in this chapter: connectivity and Hamiltonicity, distributional limit theorems, the maximal and minimal degree.

The fourth and final chapter provides an overview of the theory of random graphs from restricted classes via a *combinatorial* point of view. In particular, random graphs from minor-closed classes and bridge-addable classes are also discussed.

We would like to thank Dr. Allan Lo and Dr. Elisabetta Candellero for assisting us with the smooth running of the short course. Finally, we would like to thank the London Mathematical Society and the Engineering and Physical Sciences Research Council for their generous financial support and their invaluable assistance with various organizational matters.

1

Long paths and Hamiltonicity in random graphs

Michael Krivelevich

1 Introduction

Long paths and Hamiltonicity are certainly among the most central and researched topics of modern graph theory. It is thus only natural to expect that they will take a place of honor in the theory of random graphs. And indeed, the typical appearance of long paths and of Hamilton cycle is one of the most thoroughly studied directions in random graphs, with a great many diverse and beautiful results obtained over the past fifty or so years.

In this survey we aim to cover some of the most basic theorems about long paths and Hamilton cycles in the classical models of random graphs, such as the binomial random graph or the random graph process. By no means should this text be viewed as a comprehensive coverage of results of this type in various models of random graphs; the reader looking for breadth should rather consult research papers or a recent monograph on random graphs by Frieze and Karoński [1]. Instead, we focus on simplicity, aiming to provide accessible proofs of several classical results on the subject and showcasing the tools successfully applied recently to derive new and fairly simple proofs, such as applications of the Depth First Search (DFS) algorithm for finding long paths in random graphs and the notion of boosters.

Although this chapter is fairly self-contained mathematically, basic familiarity and hands-on experience with random graphs would certainly be of help for the prospective reader. The standard random graph theory monographs of Bollobás [2] and of Janson et al. [3] certainly provide (much more than) the desired background.

This chapter is based on a mini-course with the same name, delivered by the author at the LMS-EPSRC Summer School on Random Graphs, Geometry, and Asymptotic Structure, organized by Dan Hefetz and Nikolaos Fountoulakis at

the University of Birmingham in the summer of 2013. The author would like to thank the course organizers for inviting him to deliver the mini-course, and for encouraging him to create lecture notes for the course, which eventually served as a basis for the present chapter.

2 Tools

In this section we gather notions and tools to be applied later in the proofs. They include standard graph theoretic notation, asymptotic estimates of binomial coefficients, Chebyshev's and Chernoff's inequalities, and basic notation from random graphs (Section 2.1), as well as algorithmic and combinatorial tools – the DFS algorithm for graph exploration (Section 2.2), the so-called rotation-extension technique of Pósa, and the notion of boosters (Section 2.3).

2.1 Preliminaries

2.1.1 Notation and terminology

Our graph theoretic notation and terminology are fairly standard. In particular, for a graph $G = (V,E)$ and disjoint vertex subsets $U,W \subset V$, we denote by $N_G(U)$ the external neighborhood of U in G as: $N_G(U) = \{v \in V - U : v \text{ has a neighbor in } U\}$. The number of edges of G spanned by U is denoted by $e_G(U)$ and the number of edges of G between U and W is $e_G(U,W)$. When the graph G is clear from the context, we may omit G in the subscript in the above notation.

Path and cycle lengths are measured in edges.

When dealing with graphs on n vertices, we will customarily use N to denote the number of pairs of vertices in such graphs: $N = \binom{n}{2}$.

2.1.2 Asymptotic estimates

We will use the following, quite standard and easily proven, estimates of binomial coefficients. Let $1 \le x \le k \le n$ integers. Then

$$\left(\frac{n}{k}\right)^k \le \binom{n}{k} \le \left(\frac{en}{k}\right)^k, \tag{2.1}$$

$$\frac{\binom{n-x}{k-x}}{\binom{n}{k}} \le \left(\frac{k}{n}\right)^x, \tag{2.2}$$

$$\frac{\binom{n-x}{k}}{\binom{n}{k}} \le e^{-\frac{kx}{n}}. \tag{2.3}$$

2.1.3 Chebyshev and Chernoff

Chebyshev's inequality helps to show concentration of a random variable X around its expectation, based on the first two moments of X. It reads as follows: let X be a random variable with expectation μ and variance σ^2. Then for any $a > 0$,

$$Pr[|X-\mu| \geq a\sigma] \leq \frac{1}{a^2}.$$

The following are very standard bounds on the lower and the upper tails of the binomial distribution due to Chernoff: If $X \sim Bin(n,p)$, then

- $Pr\,(X < (1-a)np) < \exp\left(-\frac{a^2np}{2}\right)$ for every $a > 0$.
- $Pr\,(X > (1+a)np) < \exp\left(-\frac{a^2np}{3}\right)$ for every $0 < a < 1$.

Another trivial, yet useful, bound is as follows: let $X \sim Bin(n,p)$ and $k \in \mathbb{N}$. Then

$$Pr(X \geq k) \leq \left(\frac{enp}{k}\right)^k.$$

Indeed, $Pr(X \geq k) \leq \binom{n}{k}p^k \leq \left(\frac{enp}{k}\right)^k$.

2.1.4 Random graph, asymptotic notation

As usual, $G(n,p)$ denotes the probability space of graphs with vertex set $\{1,\ldots,n\} = [n]$, where every pair of distinct elements of $[n]$ is an edge of $G \sim G(n,p)$ with probability p, independently of other pairs. For $0 \leq m \leq N$, $G(n,m)$ denotes the probability space of all graphs with vertex set $[n]$ and exactly m edges, where all such graphs are equiprobable: $Pr[G] = \binom{N}{m}^{-1}$. One can expect that the probability spaces $G(n,p)$ and $G(n,m)$ have many similar features; when the corresponding parameters are appropriately tuned: $m = Np$, accurate quantitative statements are available, see [2] and [3]. This similarity frequently allows us to prove a desired property for one of the probability spaces, and then to transfer it to the other one.

We will also address briefly the model $D(n,p)$ of directed random graphs, defined as follows: the vertex set is $[n]$, and each of the $n(n-1) = 2N$ ordered pairs $(i,j), 1 \leq i \neq j \leq n$ is a directed edge of $D(n,p)$ with probability p, independently from other pairs.

We say that an event $\mathcal{E}_n$ occurs with high probability, or **whp** for brevity, in the probability space $G(n,p)$ if $\lim_{n\to\infty} Pr[G \sim G(n,p) \in \mathcal{E}_n] = 1$. (Formally, one should rather talk about a sequence of events $\{\mathcal{E}_n\}_n$ and a sequence of probability spaces $\{G(n,p)\}_n$.) This notion is defined in other (sequences of) probability spaces in a similar way.

Let $k \geq 2$ be an integer, and assume that $0 \leq p, p_1, \ldots, p_k \leq 1$ satisfy $(1-p) = \prod_{i=1}^{k}(1-p_i)$. Then the random graphs $G \sim G(n,p)$ and $G' = \bigcup_{i=1}^{k} G(n,p_i)$ have the exact same distribution. Indeed, it is obvious that each pair of vertices

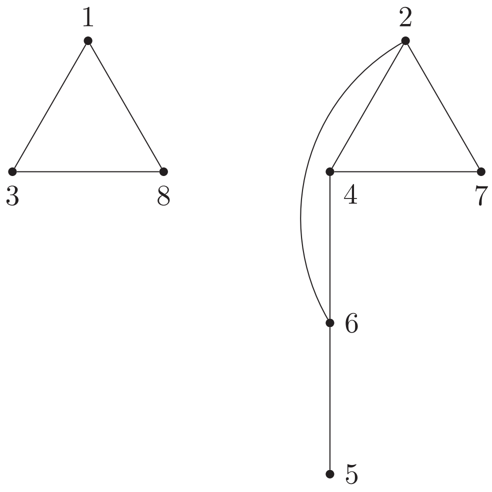
$1 \le i < j \le n$ is an edge in both graphs G, G' independently of other pairs. In G, this edge does not appear with probability $1-p$, and in order for it to not appear in G', it should not appear in any of the random graphs $G(n,p_i)$ – which happens with probability $\prod_{i=1}^{k}(1-p_i) = 1-p$, the same probability as in G. This very useful trick is called *multiple exposure* as it allows us to generate (to expose) a random graph $G \sim G(n,p)$ in stages, by generating the graphs $G(n,p_i)$ sequentially and then by taking their union. When the last probability p_k is much smaller than the rest, it is also called *sprinkling* – a typical scenario in this case is to expose the bulk of the random graph $G \sim G(n,p)$ first by generating the graphs $G(n,p_i)$, $i = 1,\dots,k-1$, to come close to a target graph property P, and then to add few random edges from the last random graph $G(n,p_k)$ (to sprinkle these few edges) to finish off the job.

2.2 Depth First Search and its applications for finding long paths

The *Depth First Search* is a well-known graph exploration algorithm, usually applied to discover connected components of an input graph. As it turns out, this algorithm is particularly suitable for finding long paths in graphs, and using it in the context of random graphs can really make wonders. We will see some of them later in the chapter.

Recall that the DFS (Depth First Search) is a graph search algorithm that visits all vertices of a (directed or undirected) graph. The algorithm receives a graph $G = (V,E)$ as an input; it is also assumed that an order π on the vertices of G is given, and the algorithm prioritizes vertices according to π. The algorithm maintains three sets of vertices, letting S be the set of vertices whose exploration is complete, T be the set of unvisited vertices, and $U = V \setminus (S \cup T)$, where the vertices of U are kept in a stack (the last-in, first-out data structure). It initializes with $S = U = \emptyset$ and $T = V$ and runs until $U \cup T = \emptyset$. At each round of the algorithm, if the set U is nonempty, the algorithm queries T for neighbors of the last vertex v that has been added to U, scanning T according to π. If v has a neighbor u in T, the algorithm deletes u from T and inserts it into U. If v does not have a neighbor in T, then v is taken out of U and is moved to S. If U is empty, the algorithm chooses the first vertex of T according to π, deletes it from T, and pushes it into U. In order to complete the exploration of the graph, whenever the sets U and T have both become empty (at this stage the connected component structure of G has already been revealed), we make the algorithm query all remaining pairs of vertices in $S = V$, not queried before. Figure 1.1 provides an illustration of applying the DFS algorithm.

Observe that the DFS algorithm starts revealing a connected component C of G at the moment the first vertex of C gets into (empty beforehand) U and completes discovering all of C when U becomes empty again. We call a



Step	S	U	T
0	$\emptyset$	$\emptyset$	$\{1,\ldots,8\}$
1	$\emptyset$	1	$\{2,\ldots,8\}$
2	$\emptyset$	1, 3	$\{2,4,\ldots,8\}$
3	$\emptyset$	1, 3, 8	$\{2,4,\ldots,7\}$
4	$\{8\}$	1, 3	$\{2,4,\ldots,7\}$
5	$\{3,8\}$	1	$\{2,4,\ldots,7\}$
6	$\{1,3,8\}$	$\emptyset$	$\{2,4,\ldots,7\}$
7	$\{1,3,8\}$	2	$\{4,5,6,7\}$
8	$\{1,3,8\}$	2, 4	$\{5,6,7\}$
9	$\{1,3,8\}$	2, 4, 6	$\{5,7\}$
10	$\{1,3,8\}$	2, 4, 6, 5	$\{7\}$
11	$\{1,3,5,8\}$	2, 4, 6	$\{7\}$
12	$\{1,3,5,6,8\}$	2, 4	$\{7\}$
13	$\{1,3,5,6,8\}$	2, 4, 7	$\emptyset$
14	$\{1,3,5,6,7,8\}$	2, 4	$\emptyset$
15	$\{1,3,4,5,6,7,8\}$	2	$\emptyset$
16	$\{1,\ldots,8\}$	$\emptyset$	$\emptyset$

Figure 1.1. Graph G with vertices labeled is on the left, and the protocol of applying the DFS algorithm to G is on the right. Observe that at any point of the algorithm execution, the set U spans a path in G.

period of time between two consecutive emptyings of U an *epoch*, each epoch corresponding to one connected component of G. During the execution of the DFS algorithm as depicted in Figure 1.1, there are two components in the graph G, and there are two epochs – the first is Steps 1–6 and the second is Steps 7–16.

The following properties of the DFS algorithm are immediate to verify:

(D1) at each round of the algorithm one vertex moves, either from T to U or from U to S;

(D2) at any stage of the algorithm, it has been revealed already that the graph G has no edges between the current set S and the current set T;

(D3) the set U always spans a path (indeed, when a vertex u is added to U, it happens because u is a neighbor of the last vertex v in U; thus, u augments the path spanned by U, of which v is the last vertex).

We now exploit the features of the DFS algorithm to derive the existence of long paths in expanding graphs.

Proposition 2.1 *Let k,l be positive integers. Assume that $G=(V,E)$ is a graph on more than k vertices, in which every vertex subset S of size $|S|=k$ satisfies $|N_G(S)|\geq l$. Then G contains a path of length l.*

Proof Run the DFS algorithm on G, with π being an arbitrary ordering of V. Look at the moment during the algorithm execution when the size of the set S of already processed vertices becomes exactly equal to k (there is such a moment as the vertices of G move into S one by one, until eventually all of them land there). By Property **(D2)** above, the current set S has no neighbors in the current set T, and thus $N(S)\subseteq U$, implying $|U|\geq l$. The last move of the algorithm was to shift a vertex from U to S, so before this move the set U was one vertex larger. The set U always spans a path in G, by Property **(D3)**. Hence G contains a path of length l. □

Proposition 2.2 *[4] Let $k<n$ be positive integers. Assume that $G=(V,E)$ is a graph on n vertices, containing an edge between any two disjoint subsets $S,T\subset V$ of size $|S|=|T|=k$. Then G contains a path of length $n-2k+1$ and a cycle of length at least $n-4k+4$.*

Proof Run the DFS algorithm on G, with π being an arbitrary ordering of V. Consider the moment during the algorithm execution when $|S|=|T|$ – there is such a moment, by Property **(D1)**. Since G has no edges between the current S and the current T by Property **(D2)**, it follows by the proposition's assumption that both sets S and T are of size at most $k-1$. This leaves us with the set U whose size satisfies: $|U|\geq n-2k+2$. Since U always spans a path by **(D3)**, we obtain a path P of desired length. To argue about a cycle, take the first and the last k vertices of P. By the proposition's assumption there is an edge between these two sets, this edge obviously closes a cycle with P, whose length is at least $n-4k+4$, as required. □

As we have hinted already, the DFS algorithm is well suited to handle directed graphs too. Similar results to those stated above can be obtained for the directed case. Here is an analog of Proposition 2.2 for directed graphs; the proof is the same, mutatis mutandis.

Proposition 2.3 *[4] Let $k<n$ be positive integers. Let $G=(V,E)$ be a directed graph on n vertices, such that for any ordered pair of disjoint subsets $S,T\subset V$ of size $|S|=|T|=k$, G has a directed edge from S to T. Then G contains a directed path of length $n-2k+1$ and a directed cycle of length at least $n-4k+4$.*

2.3 Pósa's Lemma and boosters

In this section we present yet another technique for showing the existence of long paths in graphs. This technique, introduced by Pósa in 1976 [5] in his research on Hamiltonicity of random graphs, is applicable not only for arguing about long paths, but also for various Hamiltonicity questions. And indeed, we will see its application in this context later.

In quite informal terms, Pósa's Lemma guarantees that expanding graphs not only have long paths, but also provide a very convenient backbone for augmenting a graph to a Hamiltonian one by adding new (random) edges. The fact that expanders are good for getting long paths is already not new to us – Propositions 2.1 and 2.2 are about just this. Pósa's Lemma, however, quantifies things somewhat differently, and as a result yields further benefits.

We start by defining formally the notion of an expander.

Definition 2.4 *For a positive integer k and a positive real α, a graph $G=(V,E)$ is a (k,α)-expander if $|N_G(U)|\geq\alpha|U|$ for every subset $U\subset V$ of at most k vertices.*

By the way of example, Proposition 2.1 can now be rephrased (in a somewhat weaker form – we now require the expansion of all sets of size *up to* k) as follows: if G is a (k,α)-expander, then G has a path of length at least αk. For technical reasons, Pósa's Lemma uses the particular case $\alpha=2$.

The idea behind Pósa's approach is fairly simple and natural – one can start with a (long) path P, and then, using extra edges, perform a sequence of simple deformations (rotations) until it will be possible to close the deformed path to a cycle, or to extend it by appending a vertex outside $V(P)$; then this can be repeated if necessary to create an even longer path, or to close it to a Hamilton cycle. The approach is thus called naturally the *rotation-extension* technique.