Department of Mathematics and Statistics
Memorial University of Newfoundland

Investigations into the Iterative 2-Cycle Contraction of the Directed Binomial Random Graph $\mathbb{D}(\mathbf{n}, \mathbf{p})$

Alasdair J. Graham,

under the supervision of David A. Pike

Winter 2003

Abstract. The *contraction* of the edge $e = \{u, v\}$ of the graph G is the operation which consists of replacing u and v by a single vertex whose incident edges are the edges other than e that were incident to u or v. We apply this process to 2-cycles of the binomial directed random graph $\mathbb{D}(n, p)$ and investigate the conditions under which $\left|V\left(\mathbb{D}'(n,p)\right)\right| = 1$, where $\mathbb{D}'(n, p)$ is the *iterative* 2-*cycle contraction* of $\mathbb{D}(n, p)$, that is, the graph resulting from the iterative contraction of the arcs in all 2-cycles of $\mathbb{D}(n, p)$.

Contents

1 Introduction

The web graph, the directed graph obtained by considering all world wide websites as vertices, and the hyper-links from one website to another as arcs, is increasingly the subject of scientific study. An interesting observation about this particular graph can be made, namely that it has a large subgraph inside of which there is a path from any one vertex to any other. There are other sections of the web graph that feed into this central element, and still other sections which can be reached from the central subgraph, and an important goal in web graph research deals with analysing the connectivity of these various sections. This problem motivates the discussion that follows, since various random graph models are quite useful in studying the web graph.

To this end, we look at the directed binomial random graph $\mathbb{D}(n,p)$ and examine iterative 2-cycle contractions of it. The motivation for this procedure is the examination and demarcation of various web communities, which are equivalent to strongly connected subgraphs of a randomly constructed graph. Contractions of arbitrary cycles and Hamiltonian cycles in random graph models have been previously studied and these can be used to this end, however, using 2-cycle contractions provides a finer demarcation among web communities, for the other methods can group several distinct strongly connected subgraphs as one entity, for example, if each of these strongly connected subgraphs lies on a long cycle.

Examining 2-cycle (arc) contractions in the binomial directed random graph $\mathbb{D}(n,p)$ overcomes this problem and we will investigate conditions under which the iterative contraction of the arcs of each 2-cycle in $\mathbb{D}(n,p)$ and its transformations results in the final (2-cycle-free) graph having a single vertex. Specifically, is there a connection between the threshold for strong connectivity and the threshold for the iterative 2-cycle contraction of the graph to have order one? We will consider two cases, the first for fixed probability p, the second for probability $p = p(n)$ as a function of the number of vertices n in the random digraph. A random graph generating algorithm will be presented, as well as an iterative 2-cycle contraction algorithm, and some background theory as well as some new results will be examined.

2 Preliminaries

2.1 Terminology

2.1.1 Graph Theory

A *directed graph* D consists of a set $V(D)$ of *vertices* accompanied by some subset $E(D)$ of the $n * n$ ordered pairs of (not necessarily distinct) vertices in $V(D)$. The elements of $E(D)$ are called *arcs*. To depict the arc (u,v), it is customary to draw an arrow from vertex u to vertex v, and in this case we say that u *is adjacent to* v or that v *is adjacent from* u. The arc (v,v) from vertex v to itself is called a *loop*. A *subgraph* of a graph D is a graph H such that $V(H) \subseteq V(D)$ and $E(H) \subseteq E(D)$, we say that "D contains H" and write $H \subseteq D$. A *walk* in D is a sequence of vertices $u_1, u_2, \ldots, u_n$ such that u_i is adjacent to u_{i+1} for $i \in \{1, 2, \ldots, n-1\}$. An *n-cycle* is a walk with n different vertices such that the last vertex is the first and no other vertex is repeated. A *Hamiltonian cycle* is a spanning cycle in a graph, that is, a cycle through every vertex. A *path* is a walk with no repeated vertices.

A directed graph is *strongly connected* if for any two vertices u and v there is both a path from u to v and a path from v to u. For additional background information, consult [9] and [10].

2.1.2 Probability Theory

We now pass to some probability theory. We will first describe some asymptotic notation (for a more general overview, see [6]). Given two sequences a_n and b_n, we say that $a_n = O(b_n)$ if there exist constants C and n_0 such that $|a_n| \leq Cb_n$ for all $n \geq n_0$, that is, if the sequence a_n/b_n is bounded, except possibly for some small values of n. We also say that $a_n = o(b_n)$ if $a_n/b_n \to 0$ as $n \to \infty$, that is, if for every $\varepsilon > 0$, there exists n_ε such that $|a_n| < \varepsilon b_n$ for all $n \geq n_\varepsilon$. This idea gives rise to the concept of large difference between sequences. We can represent this as $a_n \ll b_n$ or $b_n \gg a_n$ and both of these statements are true if and only if $a_n \geq 0$ and $a_n = o(b_n)$. We now pass to the definition of a probability space, of which random graphs are a specific subset.

Definition 2.1 (Bollobás [1]) A *probability space* is a triple (Ω, Σ, P) where Ω is a set, Σ is a σ-field of subsets of Ω, P is a non-negative measure on Σ, and $P(\Omega) = 1$.

2.2 Random Graph Models

A random graph model can be viewed as a probability space and an example would help to illustrate this point. We will use Erdős' original model from 1947 [5]. This is perhaps the most natural of the many undirected random graph models, and can be described as choosing one of the $2^{\binom{n}{2}}$ graphs whose vertex set is $\{1, 2, \ldots, n\}$. This is equivalent to the probability space $(\Omega, \mathcal{F}, \mathbb{P})$ where Ω is the set of all graphs with vertex set $\{1, 2, \ldots, n\}$, $\mathcal{F}$ is the class of all subsets of Ω, and each $\omega \in \Omega$ is such that

$$\mathbb{P}(\omega) = 2^{-\binom{n}{2}}.$$

Simply speaking, any graph $\omega \in \Omega$ is the result of $\binom{n}{2}$ independent tosses of a fair coin, one for each of the $\binom{n}{2}$ numbered edges in the graph, where "heads" results in the inclusion of the edge, and "tails" results in the omission of the edge.

This model is a special case of the *binomial random graph*, denoted by $\mathbb{G}(n, p)$, where p, the probability of including any edge, is a real number such that $0 \leq p \leq 1$, Ω is the set of all graphs on the n vertices $\{1, 2, \ldots, n\}$, and for any $G \in \Omega$, where $|E(G)|$ denotes the size of the edge set of G and $\mathbb{P}$ is the binomial probability on Ω,

$$\mathbb{P}(G) = p^{|E(G)|}(1-p)^{\binom{n}{2}-|E(G)|}.$$

That is, if G is a random graph obtained through the binomial random graph model, then the probability of obtaining G is equal to the product of the probabilities of obtaining the edges in G, i.e., $p^{|E(G)|}$, multiplied by the product of the probabilities of omitting the edges not in G, i.e., $(1-p)^{\binom{n}{2}-|E(G)|}$.

The model we will work with is the *directed binomial random graph*, denoted by $\mathbb{D}(n, p)$, where n is an integer and $p \in \mathbb{R}$ is such that $0 \leq p \leq 1$. This model, which is the directed counterpart to $\mathbb{G}(n,p)$, can be described as the probability space $(\Omega, \mathcal{F}, \mathbb{P})$, where Ω is the set of all directed graphs on the n vertices $\{1, 2, \dots, n\}$, $\mathcal{F}$ is the family of all subsets of Ω, and for every graph $D \in \Omega$,

$$\mathbb{P}(D) = p^{|E(D)|}(1-p)^{n^2-|E(D)|}.$$

That is, similar to its undirected counterpart, the probability of obtaining any graph $D \in \Omega$ is the product of the probabilities of obtaining the arcs in D, i.e., $p^{|E(D)|}$, multiplied by the product of the probabilities of omitting the arcs not in D, i.e., $(1-p)^{n^2-|E(D)|}$ (so we allow loops).

An example will illustrate the idea further. Suppose we want to look at graphs on $n = 4$ vertices with the probability of including each arc being $p = \frac{3}{4}$. If D^* is the directed random graph with 8 edges whose diagram is in Figure 1, then the probability that D^* will be

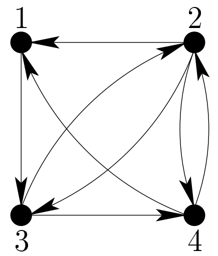

Figure 1: The directed random graph D^*

constructed in our random procedure is

$$\begin{aligned}\mathbb{P}(D^*) &= p^{|E(D^*)|}(1-p)^{n^2-|E(D^*)|} \\ &= \left(\frac{3}{4}\right)^8 \left(1-\frac{3}{4}\right)^{4^2-8} \\ &= \left(\frac{3}{4}\right)^8 \left(\frac{1}{4}\right)^8.\end{aligned}$$

Another random graph model whose directed counterpart will be essential to the discussion is the *uniform random graph*, $\mathbb{G}(n, M)$, where M is an integer such that $0 \leq M \leq \binom{n}{2}$. We take Ω as the family of all graphs on the vertex set $\{1, 2, \dots, n\}$, such that each $G \in \Omega$ has exactly M edges. Then let $\mathbb{P}$ be the uniform probability on Ω, i.e., $\mathbb{P}(G)$ is the probability of obtaining the graph G. Then, for any $G \in \Omega$,

$$\mathbb{P}(G) = \binom{\binom{n}{2}}{M}^{-1}.$$

That is, the probability of obtaining the graph G is equal to the one way to obtain the M edges in G divided by the total number of ways to obtain M edges from the $\binom{n}{2}$ possible edges.

The directed counterpart to this model, the *directed uniform random graph*, is denoted by $\mathbb{D}(n, M)$, where $0 \leq M \leq n^2$ (therefore allowing loops), and Ω is the family of all directed graphs on the vertex set $\{1, 2, \ldots, n\}$, such that each $D \in \Omega$ has exactly M arcs. Then for any $D \in \Omega$,

$$\mathbb{P}(D) = \binom{n^2}{M}^{-1}.$$

That is, similar to its undirected counterpart, the probability of obtaining the graph D is equal to the one way to obtain the M arcs in D divided by the total number of ways to obtain M arcs from the n^2 possible arcs.

Going back to our digraph D^* in Figure 1, if D^* were to be considered in the directed uniform random graph model, for $n = 4$ and $M = 8$, the probability of obtaining D^* is

$$\begin{aligned} \mathbb{P}(D^*) &= \binom{n^2}{M}^{-1} \\ &= \binom{4^2}{8}^{-1} \\ &= \frac{1}{12870}. \end{aligned}$$

A random graph, therefore, is a graph constructed by a random procedure. From standard definitions in probability theory, the "random procedure" is represented by the probability space $(\Omega, \mathcal{F}, \mathbb{P})$, and the "construction" by a function from the probability space into a suitable family of graphs.

Most of the random graph literature dealing with binomial random models is devoted to studying the probability as a function of the number of vertices of a graph. Specifically, most research has dealt with $p = p(n) \to 0$ as $n \to \infty$. Since we will consider two cases, namely, p constant and $p = p(n)$ as a function of n, we define the following.

If $\mathcal{E}_x$ is an event describing a property of a random graph depending on a parameter x, then $\mathcal{E}_x$ holds *asymptotically almost surely*, abbreviated *a.a.s.*, if $\mathbb{P}(\mathcal{E}_x) \to 1$ as $x \to \infty$.

2.3 Iterative 2-Cycle Contractions

In our study, we are interested in a series of transformations;

$$\begin{aligned} f_1 &: \mathbb{D}(n,p) \mapsto \mathbb{D}^1(n,p), \\ f_2 &: \mathbb{D}^1(n,p) \mapsto \mathbb{D}^2(n,p), \\ &\vdots \qquad \vdots \qquad \vdots \\ f_m &: \mathbb{D}^{m-1}(n,p) \mapsto \mathbb{D}^m(n,p), \end{aligned}$$

where the arcs in any 2-cycle in $\mathbb{D}^i(n,p)$ are *contracted* into a single vertex of $\mathbb{D}^{i+1}(n,p)$ and there are no 2-cycles in $\mathbb{D}^m(n,p)$. This idea of contraction is formally defined as follows:

Definition 2.2 The *contraction* of the edge $e = \{u, v\}$ of a graph G is the operation of replacing u and v by a single vertex whose incident edges are the edges other than e that were incident to u or v. A 2*-cycle contraction* is the contraction of both edges of a 2-cycle.

This definition is applicable to directed graphs so we will repeatedly contract the arcs of 2-cycles of the graph $\mathbb{D}(n,p)$, removing multiple loops and redundant arcs, until there are no 2-cycles left. This final graph will be called $\mathbb{D}'(n,p)$.

We will also attach a weight to the vertex resulting from each 2-cycle contraction in the following way. Before any contractions each vertex in the graph will have weight equal to 1. Then, the vertex resulting from each 2-cycle contraction will be assigned a weight equal to the combined weights of the two vertices just deleted. Note that the sum of the weights at any time is equal to the number of vertices in the original graph.

The reason for attaching weights is to examine the number of vertices contained in each strongly connected subgraph after all 2-cycles have been iteratively removed in the method we have just described. We will be able to do this, since in the final graph, each vertex represents a strongly connected subgraph of the original graph. If its weight is one, it was originally a single vertex, otherwise, if its weight is greater than one, the vertex represents a strongly connected subgraph of the original graph.

For example, if we go back to our graph D^*, we assign a weight of 1 to each vertex (this is the number in parentheses), and first contract the 2-cycle between vertices 2 and 3, deleting vertex 3 and reassigning a weight of 2 to vertex 2. We move the arcs to and from vertex 3 to be to and from vertex 2 and delete the second arc from vertex 2 to vertex 4 that results to produce the second graph in Figure 2. We now have a 2-cycle between vertices 1 and 2,

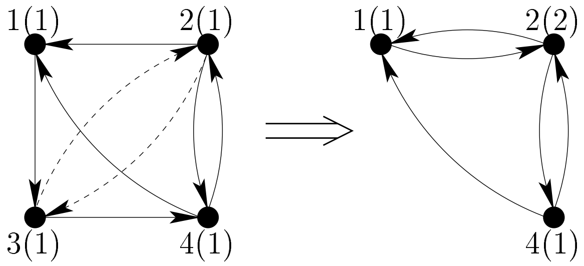

Figure 2: Contracting the 2-cycle between vertices 2 and 3

which we contract. We delete vertex 2 and reassign a weight of 3 to vertex 1 . We move the arcs to and from vertex 2 to be to and from vertex 1 and delete the second arc from 4 to 1 that results to produce the second graph in Figure 3. We now have a 2-cycle between

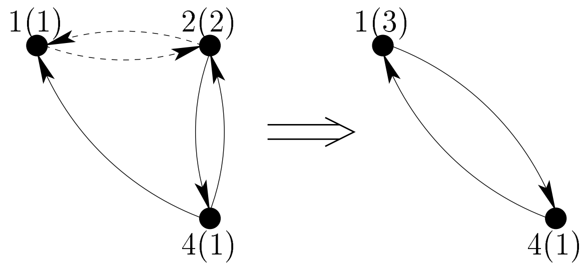

Figure 3: Contracting the 2-cycle between vertices 1 and 2

vertices 1 and 4, which we contract, reassigning a weight of 4 to vertex 1, leaving the final graph with one vertex with weight 4, as shown in Figure 4. It is important to note, as well,

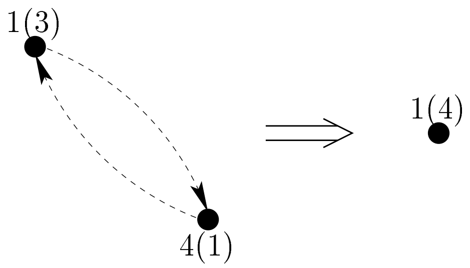

Figure 4: Contracting the 2-cycle between vertices 1 and 4

that the two arcs shown in Figure 5 do not form a 2-cycle. This follows from the definition of a directed cycle.

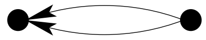

Figure 5: This is not a 2-cycle

We now show that this process of iteratively removing 2-cycles, hereafter denoted by the term *iterative* 2*-cycle contraction*, is a graph invariant. Though it may seem intuitively obvious that the iterative 2-cycle contraction of any (directed) graph is a graph invariant, for the sake of completeness, we prove this result in the case of directed graphs. Note that the invariance of the final weights of vertices follows immediately from the Lemma 2.1.

Lemma 2.1 *For any directed graph D, the iterative* 2*-cycle contraction of D is a graph invariant.*

Proof. Let D be a directed graph. Suppose that D has at least one 2-cycle (otherwise the iterative 2-cycle contraction of D is itself). There then exists, for some $l \in \mathbb{N}$, a sequence $a_l = \{a_i\}_{i=1}^{l}$ of l 2-cycle contractions that will transform D into D', a graph containing no 2-cycles. Without loss of generality, let the subsets of $V(D)$ which become contracted into single vertices be denoted $S_1, S_2, \ldots, S_j$, for some $j \in \mathbb{N}$. Without loss of generality, let vertices u and v in $V(D)$ be arbitrary and fixed inside S_1.

Suppose now that there exists some other sequence b_m of 2-cycle contractions that does not merge vertices u and v into a common final vertex. That is, if the subsets of $V(D)$ which become contracted into single vertices under b_m are denoted $T_1, T_2, \ldots, T_k$, for some $k \in \mathbb{N}$, then, for some $i \in \{1, 2, \ldots, k\}$, $u \in T_i$ and $v \notin T_i$.

Now, the two sets of vertices of D represented by the end-vertices of the 2-cycle contracted by a_1 have either been contracted at some point in the sequence b_m or they have not. If they have not, then b_m does not result in a 2-cycle-free graph since a_1 contracts a 2-cycle of arcs originally in $E(D)$. Therefore, the the 2-cycle contracted by a_1 must be contracted at some point in the sequence b_m. Now, since the 2-cycle contracted by a_1 is contracted in b_m, then the 2-cycle contracted by a_2 must occur after some number of contractions in b_m, and must also be contracted by some b_i, for otherwise, b_m does not result in a 2-cycle-free graph.

By repetition, after the sequence b_m has been performed on D, it is clear that both of the end-vertices of each 2-cycle contracted by each a_i (for $i \in \{1, 2, \ldots, l\}$) must be merged into a single vertex which represents one of the subsets $S_1, S_2, \ldots, S_j$. This results in a contradiction, namely, that u and v are not in distinct sets T_i under b_m. Therefore, any sequence of 2-cycle contractions that results in a 2-cycle-free graph must produce the subsets $S_1, S_2, \ldots, S_k$ of $V(D)$, since, by transitivity, if there is some pair of vertices which do not lie in some S_i in sequence a_n, but which are contracted in some other sequence of 2-cycle contractions, a_n will not result in a 2-cycle-free graph.

Therefore, for any two sequences of 2-cycle contractions that both result in 2-cycle-free graphs G and H, $G = H$. Hence, the 2-cycle contraction of a directed graph D is a graph invariant. □

2.4 Monotonicity and Thresholds

We call the family $\mathcal{K}$ of subsets of all graphs on n vertices *increasing* if $A \subseteq B$ and $A \in \mathcal{K}$ imply that $B \in \mathcal{K}$. Equivalently, a family of subsets is *decreasing* if its complement in the family of all graphs on n vertices is increasing. We say that a family that is either increasing or decreasing is *monotone*.

We can describe properties of graphs in this method. For example, suppose that $\mathcal{Q}$ is the property "containing a triangle". Then $\mathcal{Q}$ is an increasing property of graphs, for if A has a triangle and $A \subseteq B$, then B has a triangle as well. That is, $A \subseteq B$ and $A \in \mathcal{Q}$ imply $B \in \mathcal{Q}$.

A very interesting discovery made by Erdős and Rényi in their investigation of random graphs is the phenomenon of thresholds. For many graph properties, as $n \to \infty$, the probability that the graph possesses them jumps from 0 to 1 (or 1 to 0) with a very small change in the number of expected edges. With respect to the binomial model, we can formalise the definition as follows:

Definition 2.3 (Janson, Łuczak, and Ruciński [6]) For an increasing property $\mathcal{Q}$ of undirected graphs, a sequence $\hat{p} = \hat{p}(n)$ is called a *threshold* if

$$\mathbb{P}\big(\mathbb{G}(n,p) \in \mathcal{Q}\big) \to \begin{cases} 0 & \text{if } p \ll \hat{p}, \\ 1 & \text{if } p \gg \hat{p}. \end{cases}$$

A similar definition exists for the uniform case, with the threshold being $\hat{M} = \hat{M}(n)$. However, we need not state this formally. Instead, we will provide the directed counterpart to Definition 2.3:

Definition 2.4 For an increasing property $\mathcal{Q}$ of directed graphs, a sequence $\hat{p} = \hat{p}(n)$ is called a *threshold* if

$$\mathbb{P}\big(\mathbb{D}(n,p) \in \mathcal{Q}\big) \to \begin{cases} 0 & \text{if } p \ll \hat{p}, \\ 1 & \text{if } p \gg \hat{p}. \end{cases}$$

The property of "containing a triangle" can be used as an example of a threshold. As a special case of Theorem 3.4 of [6], we see that the threshold is $\hat{p} = 1/n$ in $\mathbb{G}(n,p)$, and $\hat{M} = n$ in $\mathbb{G}(n,M)$. That is, in $\mathbb{G}(n,p)$, a.a.s., any graph with $p = p(n) \ll 1/n$ will not contain any

3-cycle, whereas a.a.s., any graph with $p = p(n) \gg 1/n$ will contain a 3-cycle. Similarly, in $\mathbb{G}(n, M)$, a.a.s., any graph with number of edges $M = M(n) \ll n$ will not contain a 3-cycle, whereas, a.a.s., any graph with number of edges $M = M(n) \gg n$ will contain a 3-cycle (see pages 56 and 59 of [6]).

The idea of a threshold is important for us because we desire to look at the directed graph property "the iterative 2-cycle contraction has 1 vertex". This property is clearly increasing; suppose that the size of the vertex set of the iterative 2-cycle contraction of graph A is 1. Then if $A \subseteq B$, the iterative 2-cycle contraction of B must clearly have one vertex in its iterative 2-cycle contraction as well, for each of the n vertices in B must have at least one arc emanating from it and at least one arc entering it. This follows from the fact that if $a_1, a_2, \ldots, a_l$ is a sequence of 2-cycle contractions for A, resulting in 1 final vertex, it is also a sequence for B resulting in 1 final vertex.

Therefore, it makes sense to ask whether the property that we are interested in has a threshold. Theorem 1.24. of [6] provides the answer. We state it as Theorem 2.1:

Theorem 2.1 *Every monotone property has a threshold.*

The idea of a threshold for the iterative 2-cycle contraction of $\mathbb{D}(n, p)$ motivates the discussion in Subsection 4.2 for $p = p(n)$ as a function of n. However, before we pass to the results, we provide some background information and results on (strong) connectivity and Hamiltonian cycles in random graphs, as the concept of connectivity is closely related to what we are studying.

3 Contextualizing the Question

In order to give some background information on the problem, we present some previously stated results that have to do with (strong) connectivity in binomial and uniform random graph models, with an emphasis on the directed case. We will also present some results having to do with Hamiltonian and long cycles, as these results provide a good point of contrast with the method which we develop here. We will first present undirected results, and then pass to the directed results.

3.1 Undirected Results

We first state the major connectivity result for the binomial and uniform random graph models. This theorem appears in [1] where it is given an alternate proof and we state the theorem in this form (see p. 150-151 of [1]); the original appeared at least in part in [4].

Theorem 3.1 (Erdős and Rényi, 1959) *Let $\mathcal{C}$ be the property of the graph being connected, $c \in \mathbb{R}$ be fixed and let $M = \lfloor \frac{n}{2} \ln n + c + o(1) \rfloor \in \mathbb{N}$ and $p = \frac{1}{n}[\ln n + c + o(1)]$. Then, as $n \to \infty$,*

$$\mathbb{P}\big(\mathbb{G}(n, M) \in \mathcal{C}\big) \to e^{-e^{-c}}$$

and

$$\mathbb{P}\big(\mathbb{G}(n, p) \in \mathcal{C}\big) \to e^{-e^{-c}}.$$

We will see that in Subsection 3.2, a partial counterpart to this result is established, namely, in the directed uniform case. We expand the directed result to include the binomial case in Subsection 4.2. However, we first examine a result having to do with Hamiltonian cycles in the undirected case. This result stems from Bollobás' 1983 paper (see [2]) and we will state it in the form that it appears in [1] on p. 189.

Theorem 3.2 (Bollobás, 1983) *Let $\omega(n) \to \infty$, $p = \frac{1}{n}(\ln n + \ln\ln n + \omega(n))$, and $M(n) = \lfloor \frac{n}{2}(\ln n + \ln\ln n + \omega(n)) \rfloor$. Then a.a.s., $\mathbb{G}(n,p)$ is Hamiltonian, and a.a.s., $\mathbb{G}(n,M)$ is Hamiltonian.*

It is not known whether the directed counterpart to this result exists. However, if it did, it would produce a threshold for strong connectivity (since any Hamiltonian graph is obviously strongly connected), albeit a coarse one. If the result is assumed to be modified only slightly in its directed case, then it will be shown later on in Subsection 4.2 that a finer threshold can be established. We now pass to some directed results.

3.2 Directed Result

We present I. Palásti's result on strongly connected directed random uniform graphs from [8]. As mentioned already, we will extend this result to the directed binomial model in Subsection 4.2.

Theorem 3.3 (Palásti, 1966) *Let $\mathcal{C}$ be the property that $\mathbb{D}(n,M)$ is strongly connected, where $0 \le M \le n^2$ (we allow loops). Then for c an arbitrary, fixed real number, if*

$$M = M(n) = \lfloor n \ln n + cn \rfloor$$

then

$$\lim_{n\to\infty} \mathbb{P}\big(\mathbb{D}(n,M) \in \mathcal{C}\big) = e^{-2e^{-c}}.$$

From this result, we pass to the new theory, first examining the constant case for p, and then considering the case where $p = p(n)$ is a function of n.

4 New Theory

4.1 The Constant Case — $0 < p < 1$ is fixed

Initially, it may not seem obvious that for any fixed probability p, the size of the vertex set of the iterative 2-cycle contraction of $\mathbb{D}(n,p)$ is 1 a.a.s., as n approaches infinity. However, after some reflection, it becomes clear that for arbitrarily large n values, the number of edges becomes very large, even for very small fixed probabilities. From an analytical viewpoint, then, the result makes more sense. We now state and prove the desired result as:

Theorem 4.1 *For any directed binomial random graph $\mathbb{D}(n,p)$, where $0 < p < 1$ is fixed, the size of the vertex set of the iterative 2-cycle contraction of $\mathbb{D}(n,p)$ is 1 a.a.s., as $n \to \infty$.*

Proof. We use Mathematical Induction:
Let $\mathbb{D}(n,p)$ be the directed binomial random graph, where $0 < p < 1$ is fixed. Choose two vertices u and v in $V\big(\mathbb{D}(n,p)\big)$. The probability that there does not exist a 2-cycle between u and v is $1 - p^2$. Therefore, the probability that there does not exist a 2-cycle between u and any other vertex in $\mathbb{D}(n,p)$ is $(1-p^2)^{n-1}$.

Thus, the probability that there exists a 2-cycle between u and at least one other vertex in $\mathbb{D}(n,p)$ is $1-(1-p^2)^{n-1}$, and since $0 < p < 1$, we have $0 < 1 - p^2 < 1$, and so

$$\begin{aligned}\lim_{n\to\infty}\left[1-(1-p^2)^{n-1}\right] &= 1 - \lim_{n\to\infty}(1-p^2)^{n-1}\\ &= 1-0\\ &= 1.\end{aligned}$$

So the probability that we can amalgamate u with some other vertex in $\mathbb{D}(n,p)$ by means of a 2-cycle contraction is 1, a.a.s.

Now, given a subset $S \subseteq V\big(\mathbb{D}(n,p)\big)$ of m vertices whose 2-cycle contraction has one vertex, we want to show that a.a.s., there is some vertex in $V\big(\mathbb{D}(n,p)\big)\setminus S$ that shares a 2-cycle with the vertex into which the vertices of S have been merged.

The probability that there exists at least one arc from S to some fixed vertex v in $V\big(\mathbb{D}(n,p)\big)\setminus S$ is $1-(1-p)^m$. Similarly, the probability that there exists at least one arc from v to S is $1-(1-p)^m$.

Therefore, the probability that there is no 2-cycle between S and any vertex in $V\big(\mathbb{D}(n,p)\big)\setminus S$ is $\big(1-[1-(1-p)^m]^2\big)^{n-m}$ and so the probability that there exists at least one vertex in $V\big(\mathbb{D}(n,p)\big)\setminus S$ that possesses a directed arc to and from S is $1-\big(1-[1-(1-p)^m]^2\big)^{n-m}$. We have

$$\begin{aligned}0<p<1 &\implies 0<(1-p)^m<1\\ &\implies 0<[1-(1-p)^m]^2<1\\ &\implies 0<\big(1-[1-(1-p)^m]^2\big)^{n-m}<1.\end{aligned}$$

Therefore,

$$\begin{aligned}\lim_{n\to\infty}\left[\big(1-[1-(1-p)^m]^2\big)^{n-m}\right] &= 1-\lim_{n\to\infty}\big(1-[1-(1-p)^m]^2\big)^{n-m}\\ &= 1-0\\ &= 1.\end{aligned}$$

Therefore, the probability that we can add one vertex from $V\big(\mathbb{D}(n,p)\big)\setminus S$ to S without increasing the size of the iterative 2-cycle contraction of S is a.a.s. 1; that is, we can increase the size of S while ensuring that it will always reduce to one vertex with repeated 2-cycle contractions.

Hence, by Mathematical Induction, $\big|V\big(\mathbb{D}'(n,p)\big)\big| = 1$ a.a.s. as $n\to\infty$, where $\mathbb{D}'(n,p)$ is the 2-cycle contraction of $\mathbb{D}(n,p)$. $\square$

4.2 The Functional Case — $\mathbf{p} = \mathbf{p}(\mathbf{n})$

In this section, we consider $p = (n)$ as a function of n.

In order to contextualize further, we adapt Proposition 1.12. of [6] (see also [7]). This proposition has a much wider scope than we desire, for it concerns random subsets of a family of sets. The special case that we will state is for random 2-subsets of the family of all 2-subsets on n numbers, that is, for random graphs with n vertices. This being said, we state:

Proposition 1 (Łuczak [7]) *Let $\mathcal{Q}$ be an arbitrary property of subsets of the family of all graphs on n vertices with $p = p(n) \in [0,1]$, $0 \leq a \leq 1$, and $N = \binom{n}{2}$. If for every sequence $M = M(n)$ such that $M = Np + O\big(\sqrt{Np(1-p)}\big)$ it holds that $\mathbb{P}\big(\mathbb{G}(n,M) \in \mathcal{Q}\big) \to a$ as $n \to \infty$, then also $\mathbb{P}\big(\mathbb{G}(n,p) \in \mathcal{Q}\big) \to a$ as $n \to \infty$.*

We will prove the directed counterpart to this proposition.

Lemma 4.1 *Let $\mathcal{Q}$ be an arbitrary property of subsets of of the family of all directed graphs on n vertices with $p = p(n) \in [0,1]$, $0 \leq a \leq 1$, and $N = n^2$. If for every sequence $M = M(n)$ such that $M = Np + O\big(\sqrt{Np(1-p)}\big)$ it holds that $\mathbb{P}\big(\mathbb{D}(n,M) \in \mathcal{Q}\big) \to a$ as $n \to \infty$, then also $\mathbb{P}\big(\mathbb{D}(n,p) \in \mathcal{Q}\big) \to a$ as $n \to \infty$.*

Proof. Let $C \in \mathbb{R}$, $C \geq 0$, and for each n define

$$\mathcal{M}(C) = \left\{ M : |M - Np| \leq C\sqrt{Np(1-p)} \right\}.$$

Let $M_{\text{inf}} \in \mathcal{M}(C)$ be such that $\mathbb{P}\big(\mathbb{D}(n, M_{\text{inf}}) \in \mathcal{Q}\big) \leq \mathbb{P}\big(\mathbb{D}(n,M) \in \mathcal{Q}\big)$ for each $M \in \mathcal{M}(C)$. Now, if $E_p = \big|E\big(\mathbb{D}(n,p)\big)\big|$, then by the law of total probability,

$$\begin{aligned}
\mathbb{P}\big(\,\mathbb{D}(n,p) \in \mathcal{Q}\,\big) &= \sum_{M=0}^{N} \mathbb{P}\big(\,\mathbb{D}(n,p) \in \mathcal{Q} \,\big|\, E_p = M\,\big) \cdot \mathbb{P}\big(E_p = M\,\big) \\
&= \sum_{M=0}^{N} \mathbb{P}\big(\,\mathbb{D}(n,M) \in \mathcal{Q}\,\big) \cdot \mathbb{P}\big(E_p = M\,\big) \\
&\geq \sum_{M \in \mathcal{M}(C)} \mathbb{P}\big(\,\mathbb{D}(n,M_{\text{inf}}) \in \mathcal{Q}\,\big) \cdot \mathbb{P}\big(\,E_p = M\,\big) \\
&\geq \mathbb{P}\big(\,\mathbb{D}(n,M_{\text{inf}}) \in \mathcal{Q}\,\big) \cdot \mathbb{P}\big(\,E_p \in \mathcal{M}(C)\,\big).
\end{aligned}$$

Since $\mathbb{D}(n,p)$ obeys a binomial distribution, we have $\mathbb{E}(E_p) = Np$ and $\text{Var } E_p = Np(1-p)$. Therefore, using Chebyshev's Inequality and our assumption that $\mathbb{P}\big(\,\mathbb{D}(n,M_{\text{inf}}) \in \mathcal{Q}\,\big) \to a$, we have that for $t = C\sqrt{Np(1-p)}$,

$$\begin{aligned}
\mathbb{P}\big(\,|E_p - \mathbb{E}(E_p)| \geq t\,\big) &\leq \frac{\text{Var } E_p}{t^2} \\
\Longrightarrow \mathbb{P}\big(\,E_p \notin \mathcal{M}(C)\,\big) &\leq \frac{\text{Var } E_p}{\big(C\sqrt{Np(1-p)}\big)^2} = \frac{1}{C^2}.
\end{aligned}$$

Hence,

$$\liminf_{n\to\infty} \mathbb{P}\big(\,\mathbb{D}(n,p) \in \mathcal{Q}\,\big) \geq a \liminf_{n\to\infty} \mathbb{P}\big(\,E_p \in \mathcal{M}(C)\,\big) \geq a\left(1 - \frac{1}{C^2}\right).$$

Similarly, if M_{sup} maximizes $\mathbb{P}\big(\,\mathbb{D}(n,M)\in\mathcal{Q}\,\big)$ for $M\in\mathcal{M}(C)$, we have

$$\begin{aligned}\mathbb{P}\big(\,\mathbb{D}(n,p)\in\mathcal{Q}\,\big) \;\;&\leq\;\; \mathbb{P}\big(\,\mathbb{D}(n,M_{\text{sup}})\in\mathcal{Q}\,\big)\cdot\mathbb{P}\big(\,E_p\in\mathcal{M}(C)\,\big)\\ &\quad + \mathbb{P}\big(\,\mathbb{D}(n,M_{\text{sup}})\in\mathcal{Q}\,\big)\cdot\mathbb{P}\big(\,E_p\notin\mathcal{M}(C)\,\big)\end{aligned}$$

which implies that

$$\limsup_{n\to\infty}\mathbb{P}\big(\,\mathbb{D}(n,p)\in\mathcal{Q}\,\big)\leq a\left(1-\frac{1}{C^2}\right)+a\frac{1}{C^2}.$$

The result follows when we let $C\to\infty$. □

We can now incorporate I. Palásti's result on strongly connected directed random uniform graphs into the binomial model.

Theorem 4.2 *Let $\mathcal{C}$ be the property that $\mathbb{D}(n,p)$ is strongly connected. Then for c an arbitrary, fixed real number, if*

$$p=p(n)=\frac{\ln n+c}{n},$$

then

$$\lim_{n\to\infty}\mathbb{P}\big(\mathbb{D}(n,p)\in\mathcal{C}\big)=e^{-e^{-2c}}.$$

Proof. We apply Lemma 4.1 to Theorem 3.3. For this to work, we need $M=Np+O\big(\sqrt{Np(1-p)}\big)$ to hold. This will work if we let $p=p(n)$ be such that

$$M=\lfloor n\ln n+cn\rfloor=Np=n^2p.$$

Moreover, we can relax the equation to give

$$n\ln n+cn=n^2p \quad\Longrightarrow\quad p=\frac{\ln n+c}{n},$$

completing the proof. □

It now follows trivially that the function $\hat{p}=\frac{\ln n+c}{n}$ is a threshold for the strong connectivity of $\mathbb{D}(n,p)$.

Corollary 4.1 *Let $\mathcal{C}$ be the property that $\mathbb{D}(n,p)$ is strongly connected. Then for any fixed $c\in\mathbb{R}$, if $p=p(n)$ is a function of n, the following is a threshold for $\mathcal{C}$:*

$$\lim_{n\to\infty}\mathbb{P}\big(\mathbb{D}(n,p)\in\mathcal{C}\big)\to\begin{cases}0 & \textit{if } p\ll\frac{\ln n+c}{n},\\ 1 & \textit{if } p\gg\frac{\ln n+c}{n}.\end{cases}$$

Proof. Either $\hat{p}=\frac{\ln n+c}{n}$ is a threshold for $\mathcal{C}$ or it is not. If it is not, then either $\mathbb{P}\big(\mathbb{D}(n,\hat{p})\in\mathcal{C}\big)\to1$ as $n\to\infty$, a.a.s., or $\mathbb{P}\big(\mathbb{D}(n,\hat{p})\in\mathcal{C}\big)\to0$ as $n\to\infty$, a.a.s.

However, from Theorem 4.2, we know that $\mathbb{P}\big(\mathbb{D}(n,\hat{p})\in\mathcal{C}\big)\to e^{-e^{-2(c)}}$ as $n\to\infty$, and $\forall$ $c\in\mathbb{R}$, it is true that

$$0<e^{-e^{-2c}}<1.$$

Therefore, $\hat{p}=\frac{\ln n+c}{n}$ must be a threshold (for any real value of c) for $\mathcal{C}$. □

We have not yet proven any result concerning the threshold for the iterative 2-cycle contraction of $\mathbb{D}(n,p)$, yet observation of similar results in other cases yields the following conjecture (see [6] for other (undirected) results where the necessary condition is sufficient, e.g., connectivity, 1-factors, etc.):

Conjecture 4.1 *Let $\mathcal{I}$ be the property that $\mathbb{D}(n,p)$ has iterative 2-cycle contraction of order 1. Then if $p = p(n)$ is a function of n and c is a fixed real number, the following is conjectured to be a threshold for $\mathcal{I}$:*

$$\lim_{n\to\infty} \mathbb{P}\big(\mathbb{D}(n,p) \in \mathcal{I}\big) \to \begin{cases} 0 & \text{if } p \ll \frac{\ln n + c}{n}, \\ 1 & \text{if } p \gg \frac{\ln n + c}{n}. \end{cases}$$

5 Analysis

5.1 Algorithms

We will start with a short description of the **C** program that we used to build and then contract graphs randomly constructed under the directed binomial random graph model. When the program is run, the user must enter on the command line the following items:

- the number n of vertices of each graph to be built
- the minimum p-value (probability) to be used
- the total number of different p-values to be considered
- the increment between each p-value
- the number of graphs to be calculated for each p-value
- an optional seed value (if included, the program builds whatever graphs were built using that seed value previously)

Using this sequence of command line values, we are able to build graphs for any sequence of uniformly spaced p-values, e.g., we can build 100 graphs on 1000 vertices for each p-value running from $p = 0$ to $p = 0.10$ with p incrementing by $\Delta p = 0.01$ with each iteration.

After these user-inputed values have been read by the program, if there is no problem with them, it will take the initial p-value and construct however many graphs we want. Note that each graph will be constructed using the directed binomial random model. After each graph is constructed, we then determine its iterative 2-cycle contraction and record the weight of its maximally-weighted vertex.

After all the graphs for a certain p-value have been built and contracted, we then compute the maximum, minimum, and average of the set of maximum weights that we recorded, then go on to the next p-value and repeat the process.

We used two main algorithms in our program; the first one generates a user-inputed number of directed random graphs for the user-inputed range of probability p. The second algorithm iteratively contracts the 2-cycles of each graph constructed by the first algorithm. We will introduce each algorithm, and give a brief overview of some advantages and disadvantages of the second since the first is quite straightforward.

5.1.1 A Random Graph Generating Algorithm

This first algorithm is quite easy to understand. For a graph on n vertices, we first build the $n * n$ array of characters which represents the adjacency matrix of the graph and initialize each entry to zero. Then, for each arc from vertex i to vertex j (where i and j run from 1 to n inclusive), represented by the value of the $n*(i-1)+j-1^{\text{th}}$ position of the array, we obtain through use of the `drand48()` function, a randomly chosen real number, $z \in [0, 1)$. If $z \le p$, where p is the probability of choosing each arc, then we include the arc by replacing the 0 by a 1 in the arc's entry in the adjacency array. However, if $z > p$, we have not chosen the arc, so we leave the adjacency array as it is. This idea can be summarized in the following algorithm:

Algorithm 1.
Given: An adjacency matrix `A` of size $n * n$ where entry $n * (i - 1) + j - 1$ corresponds to the entry for the arc from vertex i to vertex j and is initially set to 0.

```
for ( i = 1,2,...,n)
   for ( j = 1,2,...,n)
      if ( j != i)
        {
        z = drand48()
        if ( p >= z)
          Set entry n(i - 1) + j - 1 of A to 1
        }
```

In the next section, we will discuss the more complicated iterative 2-cycle contraction procedure.

5.1.2 An Iterative 2-Cycle Contraction Algorithm

The process involved in this algorithm has been previously introduced in Subsection 2.3. We will now provide the algorithm that describes the process.

Algorithm 2.
Given: An adjacency matrix `A` of size $n * n$ where a value of 1 in the entry $n * (i - 1) + j - 1$ corresponds to an arc from vertex i to vertex j, a binary number `contracted`, initially set to 1, and an integer, `llv`, initially set to n and corresponding to the last labelled vertex.

```
while ( contracted = 1 )
     {
     contracted = 0
     for (u = 1 ; u < llv ; u ++)
        {
        for ( v = u + 1 ; v <= llv ; v ++ )
           {
           if ((there is an arc from u to v) &&
               (there is an arc from v to u))
            {
```

```
        for (w = 1 ; w <= llv ; w ++)
          if ((w != u) && (w != v))
           {
           if ((there is an arc from u to w) ||
               (there is an arc from v to w))
             {
             Set entry n(u - 1) + w - 1 of A to 1
             Set entry n(v - 1) + w - 1 of A to 0
             }
           if ((there is an arc from w to u) ||
               (there is an arc from w to v))
             {
             Set entry n(w - 1) + u - 1 of A to 1
             Set entry n(w - 1) + v - 1 of A to 0
             }
           }
        Set entry n(u - 1) + v - 1 of A to 0
        Set entry n(v - 1) + u - 1 of A to 0
        Increment the weight of vertex u by the weight of vertex v
        Set the weight of v to 0
        if (v != llv)
         Interchange the positions of the arcs of v and llv in A
        Decrement llv by 1
        Decrement v by 1
        Set contracted = 1
        }
       }
    }
  }
```

This process is somewhat more involved than Algorithm 1, so we will discuss its most important features. Firstly, once we are inside the **while** loop, we set `contracted` to 0 — once we have completed the iterative 2-cycle contraction, we set `contracted` to 1 to stay in the **while** loop. Secondly, the integer `llv` corresponds to the last labelled vertex. It is initially set to n, which is the last valid vertex number (since the vertices are numbered $1, 2, \ldots, n$). However, after we contract the first 2-cycle in the graph, in the adjacency array, we interchange the positions of the arcs to and from the deleted vertex v with the positions of the arcs to and from the vertex `llv`. We then decrement the value of `llv` by one. We do this so that when we iterate we will not look at the entry for v in our search for vertices. This interchanging of deleted vertices with undeleted vertices saves on operations because it effectively decreases the size of the adjacency array each time we delete a vertex. Finally, as can easily be seen, the algorithm is a basic search algorithm which cycles through all pairs of undeleted vertices in the graph looking for 2-cycles. If a 2-cycle is found, it is contracted, and the edges between each vertex of the 2-cycle and its neighbours are reassigned to one vertex of the 2-cycle.

One disadvantage of this method is that only relatively small n-values may be used. Since we only need zeroes and ones in our adjacency array, we can set the array to accept values from the character type. Each character uses 8 bits of information, i.e., 1 byte. If we want to allocate memory for $n * n$ characters, we can have at most $n = \sqrt{2^{32}} = 65536$ vertices, if we allocate the size of our adjacency array to be an unsigned integer. Even graphs of this size are relatively unmanageable due to the long running time of the program. However, the program is very useful in describing what happens at the threshold as will be seen in Subsection 5.2.

Other methods of contracting the 2-cycles could have been used. Specifically, we could have used a colouring algorithm which would contract the 2-cycles of the graph while colouring different strongly connected subgraphs with different colours. However, perhaps the most effective method would be the usage of linked lists, though this would require somewhat more involved programming.

5.2 Computational Results

We ran programs for $n = 100, 500, 1000, 5000$, and 10000, each generating 100 graphs with the directed binomial random model. We considered first a large range of p-values, and then, once we had established the general area of the threshold, we ran each program again, giving a more precise result. With each program, the data that we wanted to collect had to do with the weight of the vertices in $\mathbb{D}'(n,p)$.

Specifically, in each program, we determined three pieces of data for each value of p that we considered; we examined the maximum weight in each of the 100 graphs at each p-value, and found the maximum, minimum, and average of these maximum weights. We then displayed our findings in a graph. We have placed 7 graphs in the Appendix— they are listed as Figures 6, 7, 8, 9, 10, 11, and 12.

Note that Figures 9 and 10 both depict results for $n = 5000$ and Figures 11 and 12 both depict results for $n = 10000$. In each case, the initial program run produced the data in Figures 9 and 11, and a subsequent program run at a finer scale produced the data in Figures 10 and 12, which is obviously much more informative.

Each graph depicts three curves. Perhaps in Figures 6 and 7, they are most easily distinguishable. The curve representing the maximum set of weights climbs to the top of the graph very quickly, then oscillates for a while, then remains at the top. The curve representing the minimum set of weights climbs much later, oscillates, then remains at the top. The curve representing the average set of weights remains near the bottom of the graph until the maximum curve has gone to the top and then it immediately begins a gradual curve to the top of the graph, ending just before the minimum curve begins oscillating. Note that any set of weights must be bounded from beneath by 1 and from above by n. This is because every vertex in $\mathbb{D}'(n,p)$ must have weight at least 1 and no more than n.

For these graphs, the main point of interest is the width of the threshold, represented by the distance between the first upward oscillation of the maximum curve, and the last oscillation of the minimum curve. We have denoted this area as the threshold and provide for each of our n-values the corresponding threshold in Table 1. If more data points can be gathered, perhaps the data will support Conjecure 4.1 more fully. However, other interesting observations can be made. We will examine Figure 7 as it contains the most data points and

n	100	500	1000	5000	10000
Threshold	$[0.024, 0.061]$	$[0.0082, 0.0187]$	$[0.0063, 0.0111]$	$[0.0025, 0.0040]$	$[0.0017, 0.0027]$
$\frac{\ln n+c}{n}$	$0.046 + \frac{c}{100}$	$0.0124 + \frac{c}{500}$	$0.0069 + \frac{c}{1000}$	$0.0017 + \frac{c}{5000}$	$0.0009 + \frac{c}{10000}$

Table 1: Thresholds for certain n-values

most clearly exhibits the properties that we wish to explain.

First, note that the periods of oscillation in the maximum and minimum curves span a significant interval in the graph. These areas of oscillation correspond to p-values where, for each of the maximum and minimum curves, the value is alternating wildly between its minimum and maximum values. These areas are therefore areas of uncertainty, for we are unable to say, for instance in the case of the maximum curve, that it climbs from its minimum directly to its maximum, as one might expect. These areas of the threshold provide potential for additional research.

A second interesting observation is the nature of the average curve. It does not ascend rapidly, but unlike the maximum and minimum curves, it possesses a gradual rate of ascent, which is not linear, more like a portion of a **sine** curve. This facet of the threshold also provides potential for additional research.

6 Conclusion

To conclude, we will turn the focus back on the web graph. As our model contracts to one vertex in the constant case for large values of n, we know that it does not approximate the web graph (see [3]). What is needed is a more sophisticated random graph model that will closely replicate the parameters of the web graph, at least those which are necessary for discovering clustered web communities.

This being said, the iterative contraction process that was used to pare down the graph to its essentials is of much importance in this area of web graph research. Since it provides a finer sifting of websites, it will differentiate between and separate communities that would otherwise be lumped together. This is of extreme importance in the search for modelling the number, size and type of communities on the web.

Passing finally to the mathematical theory itself, some small contributions have been made. However, it is not so much the content of the theoretical contribution that is important at this level of research in the wide scope, but the discussion and examination of new ideas. There is much potential, both in the area of directed random graph research (for the number of results pales in proportion to results dealing with the undirected models), and in the more general area of world wide web research.

7 Acknowledgements

I would like to thank Dr. David Pike for the great amount of time, effort, advice, and encouragement that he put into this project. I would also like to thank all of those people who inspired me to take an interest in mathematics.

8 Appendix

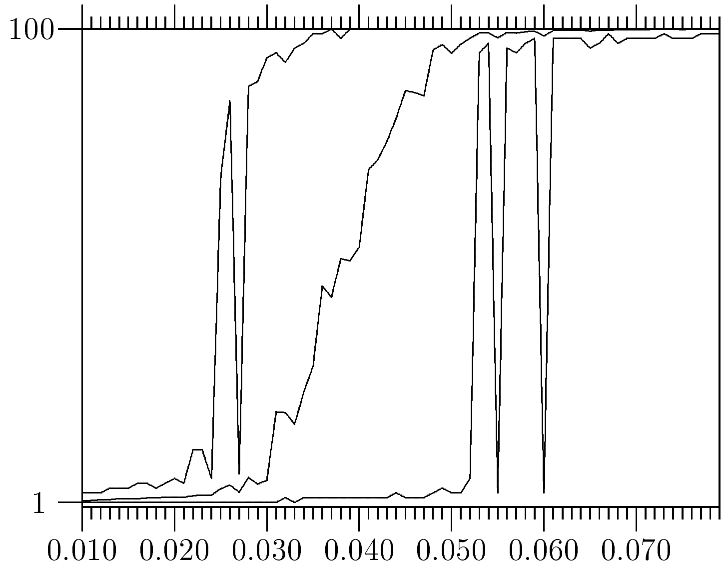

Figure 6: Maximum, minimum, and average values of the maximum weights of 100 graphs of order $n = 100$ for each p such that $0.010 \leq p \leq 0.079$ and $\Delta p = 0.001$

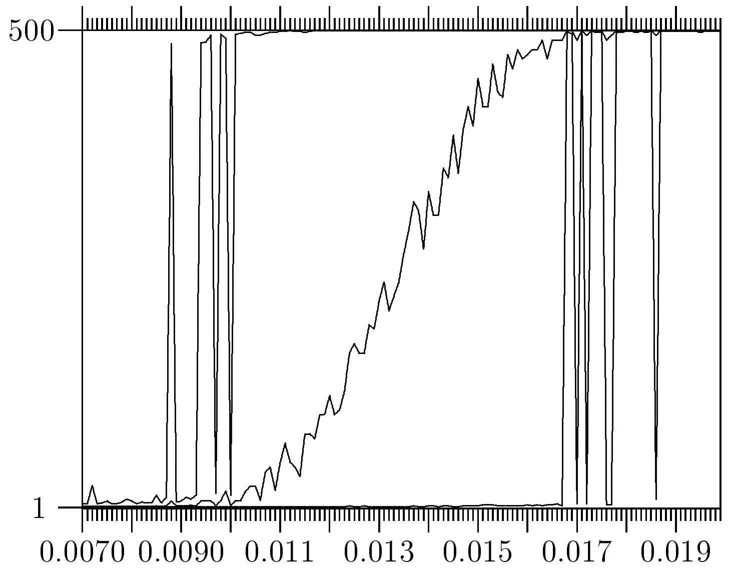

Figure 7: Maximum, minimum, and average values of the maximum weights of 100 graphs of order $n = 500$ for each p such that $0.0070 \leq p \leq 0.0199$ and $\Delta p = 0.0001$

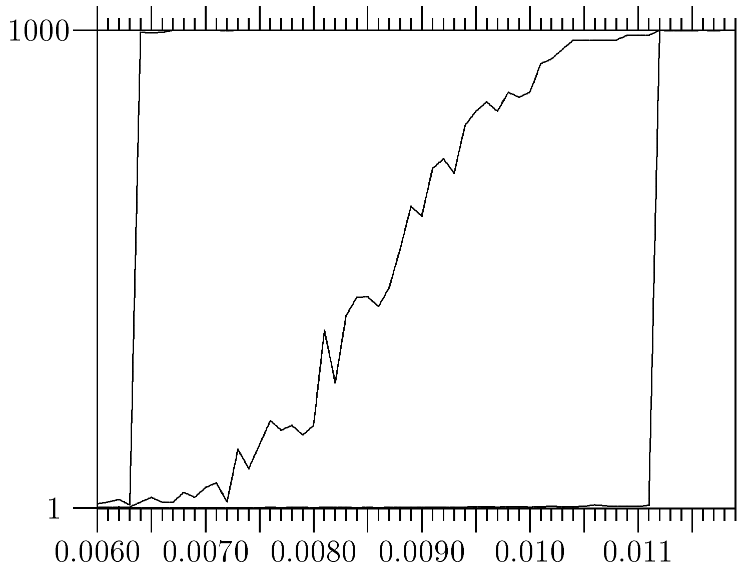

Figure 8: Maximum, minimum, and average values of the maximum weights of 100 graphs of order $n = 1000$ for each p such that $0.0060 \le p \le 0.0119$ and $\Delta p = 0.0001$

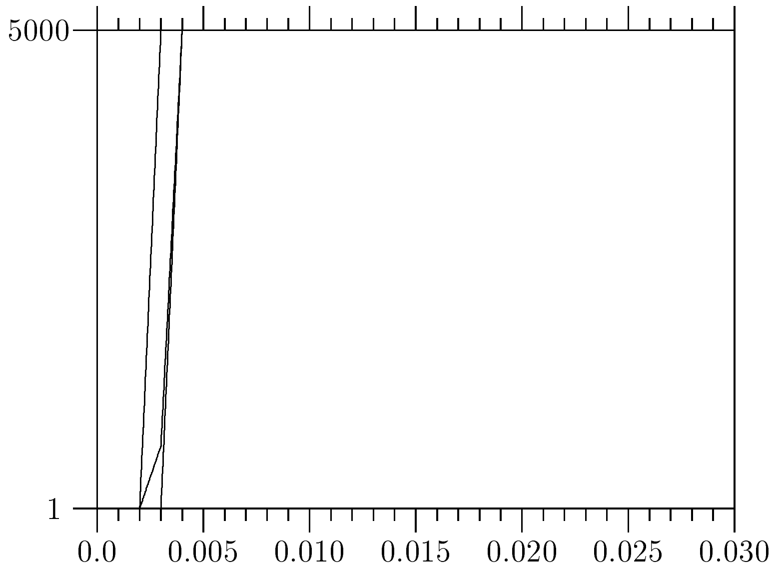

Figure 9: Maximum, minimum, and average values of the maximum weights of 100 graphs of order $n = 5000$ for each p such that $0.000 \leq p \leq 0.030$ and $\Delta p = 0.001$

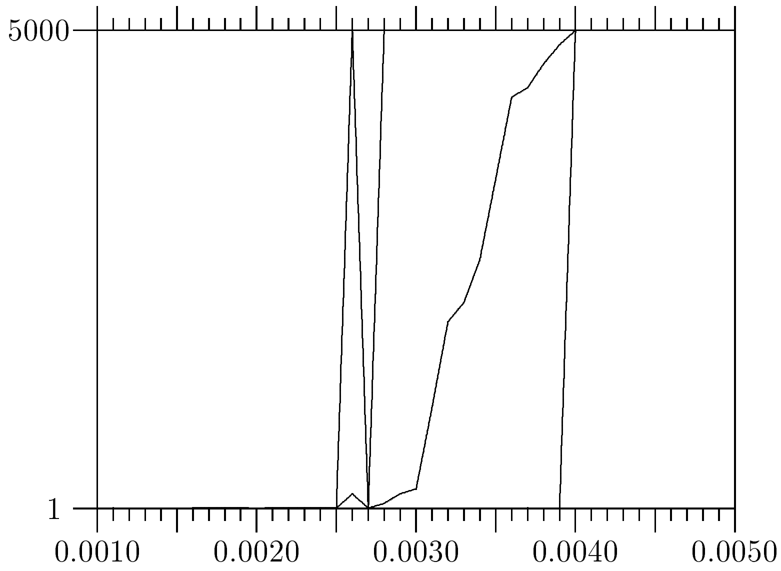

Figure 10: Maximum, minimum, and average values of the maximum weights of 100 graphs of order $n = 5000$ for each p such that $0.0010 \leq p \leq 0.0050$ and $\Delta p = 0.0001$

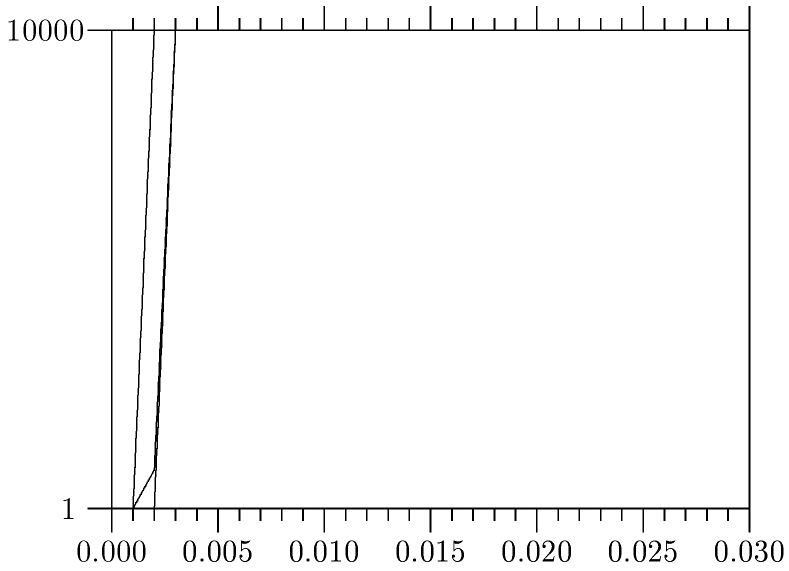

Figure 11: Maximum, minimum, and average values of the maximum weights of 100 graphs of order $n = 10000$ for each p such that $0.000 \leq p \leq 0.030$ and $\Delta p = 0.001$

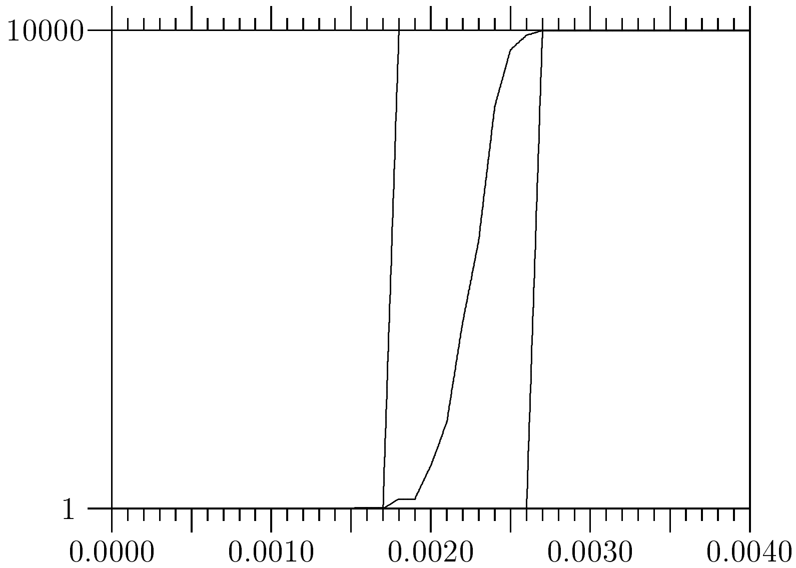

Figure 12: Maximum, minimum, and average values of the maximum weights of 100 graphs of order $n = 10000$ for each p such that $0.0000 \leq p \leq 0.0040$ and $\Delta p = 0.0001$

References

[1] B. Bollobás. *Random graphs*, Academic Press, London, England (1985).

[2] B. Bollobás. Almost all regular graphs are Hamiltonian, *Europ. J. Combinatorics.* **4** 97-106.

[3] A. Broder, R. Kumar, et al: Graph structure in the web. Proc. Of the 9th Intl. WWW Conference. ACM, (2000) 309-320.

[4] P. Erdős and A. Rényi. On random graphs **I**, *Publ. Math. Debrecen.* **6** (1959) 290-297.

[5] P. Erdős. Some remarks on the theory of graphs.*Bull. Amer. Math. Soc.* **53** (1947) 292-294.

[6] S. Janson, T. Łuczak, & A. Ruciński. *Random graphs*, John Wiley & Sons, New York, New York (2000).

[7] T. Łuczak. On the equivalence of two basic models of random graphs, *Proceedings of Random Graphs '87*, eds. M. Karoński, J. Jaworski & A. Ruciński. John Wiley & Sons, Chichester (1987) 151-158.

[8] I. Palásti. On the strong connectedness of directed random graphs, *Studia Scientiarum Mathematicarum Hungarica* **1** (1966) 205-214.

[9] E. Palmer. *Graphical evolution*, John Wiley & Sons, New York, New York (1985).

[10] D. B. West. *Introduction to graph theory*, 2nd edition, Prentice-Hall, Upper Saddle River, New Jersey (2001).