Abraham Gutiérrez Sánchez, BSc

Multi-colored Jigsaw Percolation On Random Graphs

MASTER'S THESIS

to achieve the university degree of

Master of Science

Master's degree programme: Mathematical Computer Science

submitted to

Graz University of Technology

Supervisor

Ph. D. Mihyun Kang

Institute of Discrete Mathematics

Graz, May 2017

AFFIDAVIT

I declare that I have authored this thesis independently, that I have not used other than the declared sources/resources, and that I have explicitly indicated all material which has been quoted either literally or by content from the sources used. The text document uploaded to TUGRAZonline is identical to the present master‘s thesis.

Date

Signature

Acknowledgements

I would like to thank God for giving me the opportunity to study in Austria and for crossing my path with so many wonderful people that have supported me and encouraged me in one way or another.

Contents

1 Introduction.

The field of discussion of this work is located in the theory of graphs and random graphs. We will discuss here the theory of graphs in an informal way; a more formal discussion will take place in subsections 1.1 and 1.2.

Graph Theory is the study of *graphs*. A paper [10] in the 18th century due to Leonhard Euler on the solution of the *Königsberg bridge problem* is often regarded as the first paper in this field (see [3]).

A graph can be informally described as a network of nodes and arcs, as shown in Figure 1.

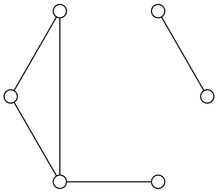


Figure 1: Example of a graph.

We will see a formal definition later. Graphs have proven to be a very powerful and important mathematical tool in a wide range of subjects such as sociology, architecture, linguistics, electrical engineering and computer sciences. We present some examples of applications of graph theory:

- (Optimal routes between two locations) We can model the streets of a city by replacing street intersections with nodes and then connect two of those nodes by arcs if and only if there is a street that goes from one street intersection to the other. In addition we can label the arcs with the length of the corresponding street. We can use this model to find shortest paths e.g. GPS systems apply special algorithms on this graph to find the shortest path between two locations.

- (Scheduling) We would like to schedule the lectures in a university so that any two lectures with a common student occur at different times. We can model this by taking a node for every lecture and an arc between two nodes if and only if two lectures share a student. If we represent the time slots by colors, the scheduling problem is the same as coloring the nodes of this graph so that no two connected nodes have the same color. Our aim is to use as few colors as possible (i.e. as few time slots as possible).

- (Social networks) We can model the "friendship network" in the world by taking a node for each person and an arc between two nodes if the people that they represent are friends. A "conjecture" called *the six degrees of separation* by Frigyes Karinthy says that in this network the "distance" between any two nodes is at most six. The "distance" between the nodes

A and B is the minimal number of “jumps” (between connected nodes) needed to go from A to B (or from B to A).

In 2006 a (scientifically non-rigorous) study by Microsoft suggested that the conjecture may be true. After checking 30 billion messages in an instant-messaging network that contained half of the global population they showed that the average distance between any two strangers was 6.6 (see [20]). This suggests that the real distance should be smaller than this value as the study traced only a small part of the actual social network.

- (Architecture) Suppose we have an $m \times n$ grid of steel beams that are attached at their ends by pin-joints. We want to make this structure rigid. We are allowed to add steel cross-beams between two pin-joints to achieve this goal. What is the minimum number of cross-beams that should be added and where they should be placed?.

  We can model the problem with a graph by taking the pin-joints as nodes and the steel beams as connections between nodes; it can be proved that *rigidity* is dependent only on the nodes and their connections. The answer to the previous question is $m + n - 1$ (see [2]).

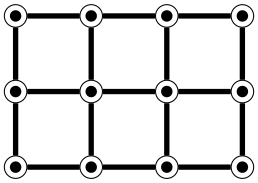


Figure 2: A 2×3 grid.

- (Sociology) The concept of *power* in sociology is a very important concept for the study of *social structures* but there is no objective definition of what power is. One way to define power is by means of graph theory: as in the third example we can take nodes for each person and arcs between two nodes if the people that they represent are acquainted with each other. The studies reveal that there is a close relation between the *potential power* a person has and the *degree* of the node that represents this person i.e. the number of acquaintances that he/she has (see [17]).

1.1 Graphs.

In this subsection we will present some basic concepts of graph theory that will be used in the thesis.

Let V be a set and let $\binom{V}{2}$:= $\{\{u, v\} : u \neq v, u \in V, v \in V\}$ i.e. the set of all unordered pairs of different elements of V. A *graph* G is an ordered pair $G = (V, E)$ where *V(G)*:= V is the set of vertices (or nodes) and $E(G)$:= $E \subseteq \binom{V}{2}$ is the set of edges. We define the *order* of G as $|G|$:= $|V|$ i.e. the number of

$\binom{V}{2}$
graph
V(G)
E(G)
order
$|G|$

vertices in G. We can see an example in Figure 3.

We say that two vertices v, w of G are *neighbors* iff $\{v, w\} \in E$. We call v and w the *end points* of the edge $\{v, w\}$. To ease notation, we use uv to denote the edge $\{u, v\}$ (note then $uv = vu$). The *degree* of a vertex v is the number of neighbors v has. For example, in the graph G_3 (see Figure 3) vertex 5 has neighbors $3, 4, 6$, i.e. degree 3.

neighbors
end points
degree

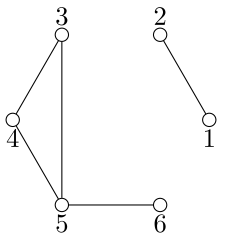


Figure 3: A graph $G_3 = (V_3, E_3)$ on vertex set $V_3 = \{1, 2, 3, 4, 5, 6\}$ and edge set $E_3 = \{12, 34, 45, 56, 35\}$. Note that $|G| = 6$ i.e. G is of order 6.

A *subgraph* of a graph $G = (V, E)$ is a graph of the form $S = (V', E')$ such that $V' \subseteq V$ and $E' \subseteq E$; we write $S \subseteq G$ to say that S is a subgraph of G. We can see an example in Figure 4.

subgraph
$S \subseteq G$

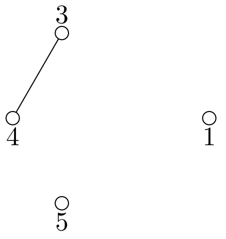


Figure 4: A subgraph $G_4 = (V_4, E_4)$ of G_3 on vertex set $V_4 = \{1, 3, 4, 5\}$ and edge set $E_4 = \{34\}$.

Given a graph $G = (V, E)$ and a subset of vertices $W \subseteq V$, we define $E[W] := E \cap \binom{W}{2}$ i.e. the edges of G whose endpoints lie in W. We call $G[W] := (W, E[W])$ the *induced subgraph* by W in G. For example, in Figure 5 we see the induced subgraph $G_3[V_4] = (\{1, 3, 4, 5\}, \{34, 45, 35\})$. Note that G_4 is not an induced subgraph of G_3.

induced subgraph

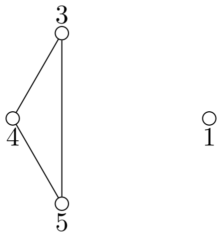


Figure 5: Induced subgraph $G_3[V_4]$.

Let $G = (V, E)$ be a graph. Given an (ordered) sequence $v_1 v_2 .... v_k$ of distinct vertices of V with $v_h v_{h+1} \in E$ for every $h \in \{1, 2, ...k - 1\}$, we define a *path* P as a subgraph of the form $(\{v_1, ..., v_k\}, \{v_1 v_2, v_2 v_3, ..., v_{k-1} v_k\})$. We call v_1 and v_k the *endpoints* of P and we say that P is a *path between* v_1 and v_k, we also say that P is of *length* $k - 1$. The *distance* between two vertices v and w is the length of a minimal path between them; if no path between them exists we say that their distance is infinite. Given an (ordered) sequence $v_1 v_2 .... v_k$ of distinct vertices of V where $v_i v_{i+1} \in E$ for every $i \in [k]$ and where we set $v_{k+1} = v_1$, we define a *cycle* of G as a subgraph of the form $\{\{v_1, ..., v_k\}, \{v_1 v_2, v_2 v_3, ..., v_k v_1\}\}$. If a cycle C in G contains all vertices of G, we call C a *Hamilton cycle*. For convenience of notation we will identify path and cycles with sequences of vertices. However these sequences are not uniquely defined: for example $v_1 v_2 ... v_k v_1$ represents the same cycle as $v_1 v_k ... v_2 v_1$ since we do not take into account the orientation of the cycle. Some examples are shown in Figure 6.

path
endpoints
length
distance
cycle
Hamilton cycle

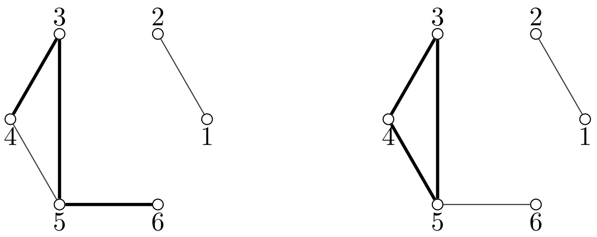


Figure 6: The figure on the left highlights the path 4356 in G_3. This is a path between vertices 4 and 6. The figure on the right highlights the cycle 345 in G_3. This graph does not contain a Hamilton cycle. The distance between vertices 4 and 6 is 2; the distance between vertices 1 and 6 is infinite because there is no path between them.

A graph $G = (V, E)$ is *connected* iff there is a path between every two distinct vertices of V. A graph that is not connected is called *disconnected*. A *connected component* of a graph $G = (V, E)$ is an induced subgraph $G[W]$ that is connected and maximal in the sense that there is no connected induced subgraph of G that properly contains $G[W]$. For example, G_3 is disconnected with connected components $G[\{1, 2\}], G[\{3, 4, 5, 6\}]$.

connected
disconnected
connected component

A *tree* is a connected graph which contains no cycles. Given a graph $G =$

tree

(V,E), we call a subgraph $T=(V_1,E_1)$ a *spanning tree* of G iff T is a tree and $V_1=V$. Connectedness and spanning trees are closely related, as shown in Proposition 1. This proposition tells us in particular that G_3 cannot have a spanning tree because it is not connected. We can see an example of a spanning tree in Figure 7.

spanning tree

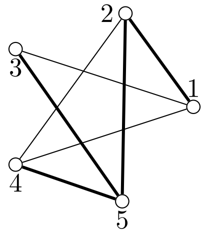


Figure 7: Spanning tree $T_7 = (\{1,2,3,4,5\},\{54,53,52,21\})$ of the graph $G_7 = \{\{1,2,3,4,5\},\{12,13,14,24,25,35,45\}\}$.

Proposition 1 (see e.g. [9]). *A graph G is connected iff G contains a spanning tree.*

A *graph property* $\mathcal{P}$ is a family of graphs. For example, the property of being connected corresponds to the family of all connected graphs. Informally we say that a graph G *has* the property $\mathcal{P}$ or *satisfies* the property $\mathcal{P}$ if $G \in \mathcal{P}$. We say that a property $\mathcal{P}$ is *monotone increasing* if given two graphs G_1 and G_2 on the same vertex set, if $G_1 \subseteq G_2$ and $G_1 \in \mathcal{P}$ then $G_2 \in \mathcal{P}$. Equivalently we can describe it as a property that is not destroyed by adding edges.

graph property

monotone increasing

1.2 Random Graphs.

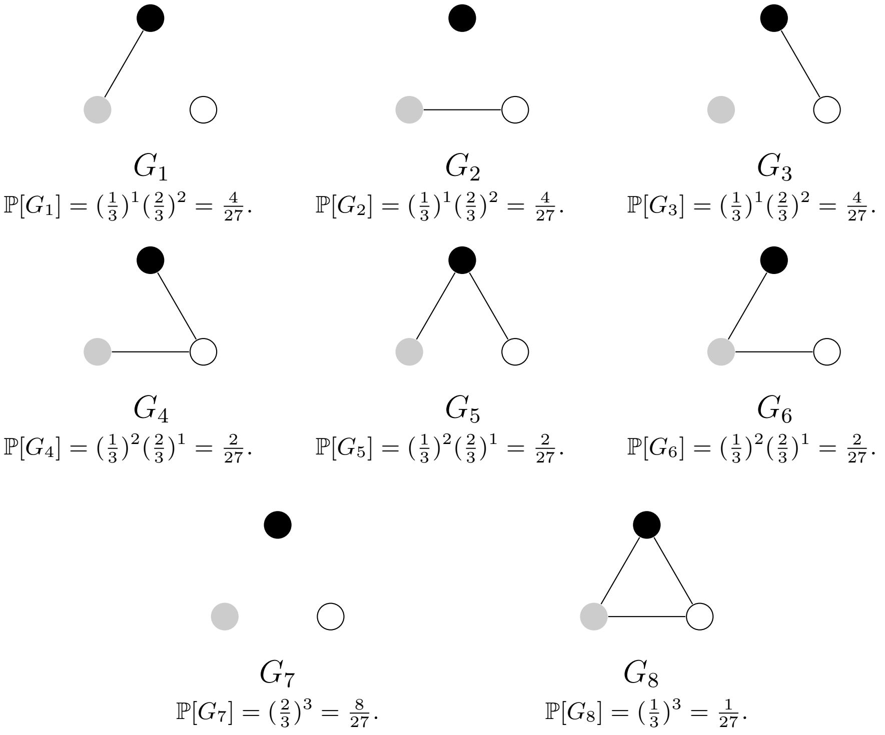
In the late 1950s and early 1960s Erdős and Rényi began to study graphs that are generated randomly. The focus is not what happens for *all* graphs but what typically happens for *most* graphs.

Random graphs are useful because they have many (though not all) of the same properties as *complex networks*, but analyzing the networks themselves may be challenging: the networks can be too large or the information about them can be incomplete; examples of complex networks include social networks, brain networks and computer networks.

Let $n \in \mathbb{N}$ and $0 \le p \le 1$. Then $\mathcal{G}(n,p)$ is the probability space of graphs on vertex set $[n]=\{1,2,...,n\}$ with probability distribution defined by $\mathbb{P}[([n],E)] = p^{|E|}(1-p)^{\binom{[n]}{2}-|E|}$ for any $E \subseteq \binom{[n]}{2}$, i.e. given a graph $G=([n],E)$, the probability that G is "selected" is $p^{|E|}(1-p)^{\binom{[n]}{2}-|E|}$. An Erdős-Rényi *binomial random graph* is a random element of $\mathcal{G}(n,p)$ usually denoted as $G(n,p)$. Alternatively (and equivalently), we can describe $G(n,p)$ as a graph on vertex set $[n]$ where each edge is present with probability p independently of each other.

$\mathcal{G}(n,p)$

binomial random graph



G_1
$\mathbb{P}[G_1] = (\frac{1}{3})^1(\frac{2}{3})^2 = \frac{4}{27}$.

G_2
$\mathbb{P}[G_2] = (\frac{1}{3})^1(\frac{2}{3})^2 = \frac{4}{27}$.

G_3
$\mathbb{P}[G_3] = (\frac{1}{3})^1(\frac{2}{3})^2 = \frac{4}{27}$.

G_4
$\mathbb{P}[G_4] = (\frac{1}{3})^2(\frac{2}{3})^1 = \frac{2}{27}$.

G_5
$\mathbb{P}[G_5] = (\frac{1}{3})^2(\frac{2}{3})^1 = \frac{2}{27}$.

G_6
$\mathbb{P}[G_6] = (\frac{1}{3})^2(\frac{2}{3})^1 = \frac{2}{27}$.

G_7
$\mathbb{P}[G_7] = (\frac{2}{3})^3 = \frac{8}{27}$.

G_8
$\mathbb{P}[G_8] = (\frac{1}{3})^3 = \frac{1}{27}$.

Figure 8: Graphs on 3 vertices with their respective probabilities in the probability space $\mathcal{G}(3, 1/3)$.

uniform model

Note that this is not quite the model that Erdős and Rényi considered; the original model used by Erdős and Rényi is known as the *uniform model*: $\mathcal{G}(n, m)$ is the probability space of all graph $H = (V, E)$ on vertex set $[n]$ with exactly $0 \leq m \leq \binom{n}{2}$ edges, and with probability distribution $\mathbb{P}[H] = 1/\binom{\binom{n}{2}}{m}$ i.e. every graph in the probability space has the same probability of being "selected", meaning the probability is *uniform*. $G(n, m)$ denotes an element chosen at random in $\mathcal{G}(n, m)$.

The binomial model is usually easier to analyse because we have independence between edges. However in some cases it is much more convenient to use the uniform model. For example, when adding edges one by one and determining a "hitting time" i.e. when the graph first satisfies a certain property.

Consider the models $\mathcal{G}(3, p = 1/3)$ and $\mathcal{G}(3, m = 1)$. In Figure 8 we see that the only graphs with non-zero probability of being "selected" in $\mathcal{G}(3, m = 1)$ are G_1, G_2 and G_3. On the other hand the probability of "selecting" one of these three graphs in $\mathcal{G}(3, p = 1/3)$ is $12/27 < 1/2$. Clearly these spaces are very different (and no non-trivial choice of p and m will change this fact). However, when $n \to \infty$ (which is what usually interest us) the two models become closely related (under some reasonable assumptions) in the sense that events that are probable in one model are also probable in the other. We can informally express this as follows:

- For any graph property $\mathcal{P}$, if for any m such that $m \sim \binom{n}{2}p$ (i.e. the number of edges is $G(n,m)$ is approximately the same as the expected number of edges in $G(n,p)$) it holds that $\mathbb{P}[G(n,m) \in \mathcal{P}] \xrightarrow{n\to\infty} p_0$ for some constant p_0, then $\mathbb{P}[G(n,p) \in \mathcal{P}] \xrightarrow{n\to\infty} p_0$.
- For a monotone graph property $\mathcal{P}$, if $\mathbb{P}[G(n,p) \in \mathcal{P}] \xrightarrow{n\to\infty} p_0$ for every $p \sim \frac{m}{\binom{n}{2}}$ then $\mathbb{P}[G(n, m = \lfloor p\binom{n}{2}\rfloor) \in \mathcal{P})] \xrightarrow{n\to\infty} p_0$.

The exact statements are expressed in Theorem 1.

Theorem 1. *(see e.g. [14]) Let $0 \leq p_0 \leq 1$, let $p = p(n)$ and $s(n) = n\sqrt{p(1-p)} \xrightarrow{n\to\infty} \infty$, and let $\omega(n) \xrightarrow{n\to\infty} \infty$. Then,*

- *If $\mathcal{P}$ is any graph property and for all $m \in \mathbb{N}$ such that $|m - \binom{n}{2}p| < \omega(n)s(n)$ it holds that $\mathbb{P}[G(n,m) \in \mathbb{P}] \xrightarrow{n\to\infty} p_0$, then $\mathbb{P}[G(n,p) \in \mathcal{P}] \xrightarrow{n\to\infty} p_0$.*
- *Let $p_- := p - \frac{\omega(n)s(n)}{n^3}$ and $p_+ := p + \frac{\omega(n)s(n)}{n^3}$. If $\mathcal{P}$ is a monotone graph property and:*
  - $\mathbb{P}[G(n,p_-) \in \mathcal{P}] \xrightarrow{n\to\infty} p_0$,
  - $\mathbb{P}[G(n,p_+) \in \mathcal{P}] \xrightarrow{n\to\infty} p_0$,

  then $\mathbb{P}[G(n, m = \lfloor p\binom{n}{2}\rfloor) \in \mathcal{P})] \xrightarrow{n\to\infty} p_0$.

This theorem means that the uniform model and the binomial model are very similar. We will only consider the binomial model in the rest of the thesis.

We will use the following *asymptotics* throughout the thesis. Given two functions $f, g : \mathbb{N} \to \mathbb{R}$ we write $g(n) = O(f(n))$ iff $|g(n)/f(n)|$ is bounded for sufficiently large n, i.e. there exist n_0 and C such that for every $n \geq n_0$, $|g(n)/f(n)| \leq C$, on the other hand we write $f(n) =\Omega(g(n))$ iff $g(n) = O(f(n))$. Similarly, we write $g(n) = o(f(n))$ (or $g \ll f$) iff $g(n)/f(n) \xrightarrow{n\to\infty} 0$, on the other hand we say that $f(n) = \omega(g(n))$ iff $g(n) = o(f(n))$. We write $f(n) = \Theta(g(n))$ iff $f(n) = O(g(n))$ and $g(n) = O(f(n))$.

asymptotics
$O(f(n))$
$\Omega(g(n))$
$o(f(n))$
$\omega(g(n))$
$\Theta(g(n))$

Given $p = p(n)$ we say that a property $\mathcal{P}$ holds *with high probability* iff $\mathbb{P}[G(n,p(n)) \in P] \xrightarrow{n\to\infty} 1$. The phrase "with high probability" will be abbreviated to *whp*.

with high probability
whp

Erdős and Rényi proved some remarkable results about random graphs including the *giant component phase transition* (Theorem 6) and the *connectedness phase transition*(Theorem 3). These are examples of a phenomenon called *phase transition*; this phenomenon refers to the radical change of "behavior" of random graphs given a small change in the probability p (or the number of edges because this is $\sim \binom{n}{2}p$).

phase transition

In general, we want to study random graphs $G(n,p)$ where $n \to \infty$ and $p(n)$ is a function of n. In this case we want to find a *threshold function* $t(n)$ for a specific property $\mathcal{P}$. i.e. a function such that:

threshold function

- If $p \ll t$, then whp $G(n, p(n))$ does *not* have the property $\mathcal{P}$;
- if $p \gg t$, then whp $G(n, p(n))$ has the property $\mathcal{P}$,

or vice versa.

In the case of the *phase transition* results of Erdős and Rényi mentioned above we not only find a threshold but a *sharp* threshold. A function $s(n)$ is a *sharp threshold function* for a property $\mathcal{P}$ iff for every $\epsilon > 0$:

sharp threshold function

- If $p(n)/s(n) \leq 1 - \epsilon$, then whp $G(n, p(n))$ does *not* have the property $\mathcal{P}$;
- if $p(n)/s(n) \geq 1 + \epsilon$, then whp $G(n, p(n))$ has the property $\mathcal{P}$,

or vice versa,

We mentioned that we would like to find thresholds but it is not clear why they should exist. It turns out that if the property we are analyzing is monotone then there is always a sharp threshold:

Theorem 2. *(E. Friedgut, G. Kalai, [13]) Every monotone graph property has a sharp threshold in $\mathcal{G}(n,p)$.*

Consider $G(n,p)$ with $p = 0.9$ and $n \to \infty$, in this case it is very probable that $G(n,p)$ will be connected (whp all vertices have $\sim 0.9n$ neighbors so any two share a common neighbor). Indeed, for any constant $0 < p \leq 1$ whp $G(n,p)$ is connected. We can argue this by calculating the probability that two fixed vertices are at distance at least 3 and then multiplying this value by $\binom{n}{2}$ to get an upper bound on the probability that there exist any two vertices at distance at least 3 in $G(n,p)$; it is easily verifiable that this upper bound tends to 0 as n tends to infinity. If we let $p \xrightarrow{n\to\infty} 0$, how does this affect the probability that $G(n,p)$ is connected? If p tends to 0 very fast (for example by letting $p = 0$) it is clear that the probability for connectedness tends to zero very fast also. The surprising fact is that if we let p tend to zero slowly enough, the probability for connectedness tends to 1; furthermore there is a threshold for this. This phenomenon is known as the *phase transition for connectedness* and was discovered by Erdős and Rényi (see [12]).

Theorem 3 (Phase transition for connectedness, [11])**.** *Let $a \geq 0$ and $G(n,p)$ a random element of $\mathcal{G}(n,p)$ where $p = a \log n/n$, then:*

- *If $a < 1$, then whp $G(n,p)$ is* not *connected.*
- *If $a > 1$, then whp $G(n,p)$ is connected.*

Remark 4. Note that as mentioned earlier, Theorem 3 (and later also Theorem 6) is not stated originally in this form by Erdős and Rényi since they used the uniform model. However, this new form of the theorem follows easily from the original statement.

Remark 5. Stronger versions of Theorem 3 are known, see e.g. [12].

A *giant component* is a connected component of a graph $G = (V, E)$ that contains a constant fraction of V. Note that this is automatically an asymptotic condition. The *giant component phase transition* theorem states that the threshold for the existence of a giant component is located at $p = 1/n$. Theorem 6 can be informally stated as saying that whp below this p all connected components are small and above this p whp there is a unique giant component and the rest are small.

giant component

Theorem 6 (Phase transition for giant component, [11])**.** *Let $c > 0$, $p = p(n) := c/n$ and $G(n,p)$ be a random element of $\mathcal{G}(n,p)$. Let C_1 the largest connected component of $G(n,p)$ and C_2 be the second-largest connected component of $G(n,p)$. Then the following holds:*

- *If $c < 1$, then whp $|C_1| = O(\ln n)$.*
- *If $c = 1$, then there is a constant $\kappa > 0$ such that for all $a > 0$,*

$$\lim_{n\to\infty} \mathbb{P}\left[|C_1| \geq an^{2/3}\right] \leq \frac{\kappa}{a^2}.$$

- *If $c > 1$, then whp there is a unique giant component and $|C_2| = O(\ln n)$.*

Remark 7*.* Stronger versions of Theorem 6 are known, see e.g. [4].

1.3 Jigsaw Percolation.

The study of the interactions between two (or more) objects (of various natures) under specific rules plays a key role in many sciences; in graph theory, interactions between graphs can also be studied. We would like to answer this question: when are two graphs on the same vertex set "jointly connected"?[1] We would like to find a meaningful interaction between two graphs so we can call them "jointly connected". One such interaction is provided by the jigsaw percolation process.

This was introduced by Brummitt, Chatterjee, Dey, and Sivakoff in [7] to model how a social network can solve a puzzle collectively. It studies the interaction between the graph of "social interactions" and the graph of "information exchange". We will see later that connectedness is a necessary (but not sufficient) condition for two graphs to complete the jigsaw percolation process successfully.

The study of how social networks solve problems has been of increasing interest. In a 2012 Scientific American article entitled "Social Network Size Linked to Brain Size" (see [18]) it is actually claimed that our brains are wired to solve difficult problems by means of social networks. We quote: "...Our brains are not as large as they are in order to provide each of us with the raw computational power to think our way out of a sticky situation, instead our brain size helps each of us to deal with the large and complex network of relationships we rely on to thrive...", the evidence referred to is [19]. Based on

[1] This term is being used rather informally and it can have a different meaning in some literature.

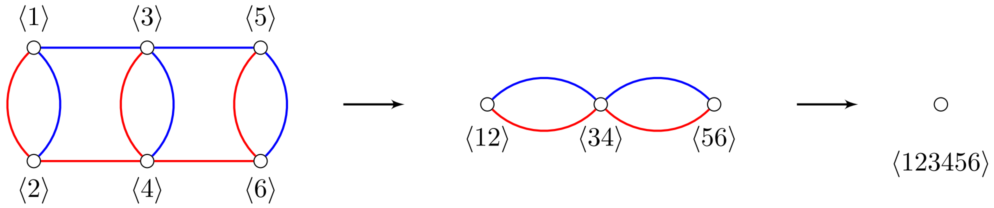


Figure 9: Percolating pair of graphs. We begin the process with a graph on vertex set $\{1, 2, 3, 4, 5, 6\}$. Each of these vertices forms a cluster by itself at the beginning. In the next step we see that the clusters $\langle 1\rangle$ and $\langle 2\rangle$ merge into a cluster $\langle 12\rangle$ because they share edges of both colors. Similarly clusters $\langle 34\rangle$ and $\langle 56\rangle$ are formed from clusters $\langle 3\rangle, \langle 4\rangle, \langle 5\rangle$ and $\langle 6\rangle$. In the final step we note that clusters $\langle 12\rangle, \langle 34\rangle$ and $\langle 56\rangle$ merge because they form a connected component in the graph of double edges.

this we believe that many models are yet to come in the study of how social networks solve problems.

In this thesis we will consider jigsaw percolation, which is a discrete time process that merges clusters of vertices according to a deterministic rule. Each of the n people in the social network has a piece of the puzzle, and the pieces of the puzzle must be combined in a certain way in order to solve it. We can model the n people by n vertices and within those n vertices we introduce a red-edged graph and a blue-edged graph. In [7] these graphs are called the *people-graph* and the *puzzle-graph* respectively. The first graph tells us whether two people are acquainted and the second whether their respective puzzle pieces are compatible. If two people v_1 and v_2 are acquainted and their puzzle pieces are compatible we merge them into one cluster v_{12} that inherits all the neighbors that v_1 and v_2 had in the red-edged and blue-edged graphs (in the formal definition we may merge connected components of clusters instead of just pairs). We go on as long as there are clusters that share a red and blue edge. If we get a single cluster at the end of the process, this means that the puzzle was solved, and we say that the process percolates or that the graphs percolate. We can see an example of this process in Figure 9. A formal definition of this process is given in Algorithm 1.

people-graph
puzzle-graph

The process cannot percolate if one of the graphs is not connected, as we can observe in Figure 10. We note that the property that two graphs percolate is a *monotone increasing property* i.e. if two graphs percolate, then adding edges into either one will not change this. This implies that if $p_1 \leq p_1'$ and $p_2 \leq p_2'$ then $\mathbb{P}\left[(G(n,p_1), G(n,p_2)) \text{ percolates}\right] \leq \mathbb{P}\left[(G(n,p_1'), G(n,p_2')) \text{ percolates}\right]$ (a formal proof of this fact is given in Proposition 7).

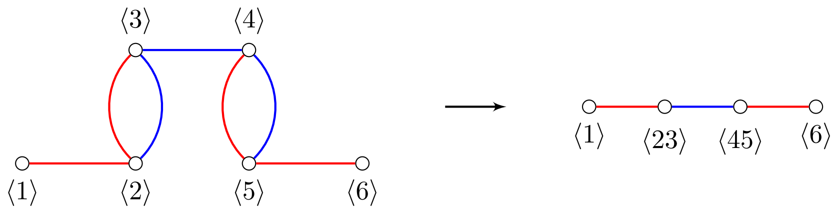


Figure 10: Non-percolating 2-fold graph. In the first step we have a graph on vertex set $\{1, 2, 3, 4, 5, 6\}$. At the beginning, these vertices form clusters by themselves. In the next step clusters $\langle 2\rangle, \langle 3\rangle$ and $\langle 4\rangle, \langle 5\rangle$ merge to form clusters $\langle 23\rangle$ and $\langle 45\rangle$. We then stop because there are no more clusters that can be merged. We are left with 4 clusters in the end, so the process does not percolate.

In [7], Brummitt, Chatterjee, Dey and Sivakoff studied the jigsaw percolation process with the people-graph as a binomial random graph and puzzle-graph being a Hamilton cycle, or some other connected graph of bounded maximum degree. Their results give upper and lower bounds for the percolation threshold probabilities.

We will now introduce some notation to state the main results of [7]. For a connected puzzle-graph and a people-graph given by a binomial random graph $G(n, p)$, we denote the probability that the puzzle is solved i.e. the two graphs percolate by $\mathbb{P}_p[Solve]$ for $0 \leq p \leq 1$ (as a function of n). This function is a polynomial in p with degree at most $\binom{n}{2}$ (as remarked in [7]). Since $\mathbb{P}_p[Solve]$ is continuous and $\mathbb{P}_0[Solve] = 0$ and $\mathbb{P}_1[Solve] = 1$, by the *intermediate value theorem* for each n there is a value $p_c(n)$ (which is unique since the function is strictly monotone) such that $\mathbb{P}_{p_c}[Solve] = 1/2$. The value $p_c(n)$ is called the *critical value* in [7].

$p_c(n)$

critical value

In [7] the following remarks are made:

- There is nothing special about the value $1/2$, which could be replaced by any positive constant c with $0 < c < 1$.
- The *critical value* $p_c(n)$ depends on the puzzle-graph but the notation does not include this dependency.

Theorem 8 (Ring puzzle, [7])**.** *If the people graph is the binomial random graph and the puzzle graph is the n-cycle, then:*

$$\frac{1}{27 \log n} \leq p_c(n) \leq \frac{\pi^2}{6 \log n}(1 + o(1)).$$

Moreover, for $p_n = \lambda / \log n$:

- *if $\lambda < 1/27$, then $\mathbb{P}_{p_n}[Solve] \xrightarrow{n\to\infty} 0$,*
- *if $\lambda > \pi^2/6$, then $\mathbb{P}_{p_n}[Solve] \xrightarrow{n\to\infty} 1$.*

In [7] the following conditions for percolation of puzzle-graphs with bounded maximum degree are obtained.

Theorem 9 (Connected puzzle of bounded degree, [7])**.** *For a binomial random people-graph solving a connected puzzle with bounded maximum degree,* $p_c(n) = O(1/\log n)$ *and* $p_c(n) = \omega(1/n^b)$ *for any* $b > 0$*. In particular, we have:*

- *If* $b > 0$ *and* $p_n = O(1/n^b)$*, then* $\mathbb{P}_{p_n}[Solve] \to 0$,
- *If* $\lambda > \pi^2/6$ *and* $p_n = \lambda/\log n$*, then* $\mathbb{P}_{p_n}[Solve] \to 1$.

The upper bounds of these theorems are proven in [7] for a general connected puzzle-graph. Under the upper-bound assumptions it is proved that whp there exists a sufficiently large vertex-subset such that the subgraphs induced by it in the people-graph and the puzzle-graph percolate; once this sufficiently large vertex-subset exists it follows that whp the people-graph and the puzzle-graph percolate.

The lower bound in the case of the ring-puzzle is proved by showing that whp the Hamilton cycle can be cut into vertex-subsets that will not merge with the people-graph. The strategy to prove the lower bound in the case of Theorem 9 is to prove inductively (on the number of steps for the percolation process) that whp the number of clusters that merge (in the previous step of the percolation process) decreases and the maximum number of vertices in any cluster is whp bounded by a constant and is therefore less than n for sufficiently large n.

In [7] it is observed that the *critical value* $p_c(n)$ marks the phase transition for the people-graph at which it begins to solve the puzzle whp; it is also observed that even though $p_c(n) \xrightarrow{n\to\infty} 0$ the *critical average degree* of a vertex $np_c(n) \xrightarrow{n\to\infty} \infty$ i.e. the average number of interactions per person in the people-graph must increase. Thus the number of interactions per person in a social network must increase as the puzzle that must be solved "grows".

In [6] Bollobás, Riordan, Slivken, and Smith replaced the people-graph and the puzzle-graph with binomial random graphs $G(n, p_1), G(n, p_2)$. Their result is a percolation threshold in terms of the product $p_1 p_2$.

Theorem 10 (see [6])**.** *There exists a constant* c *such that the following holds: for* $G_1 = G(n, p_1), G_2 = G(n, p_2)$ *independent binomial random graphs where* $0 \leq p_1 = p_1(n), p_2 = p_2(n) \leq 1$*. Then:*

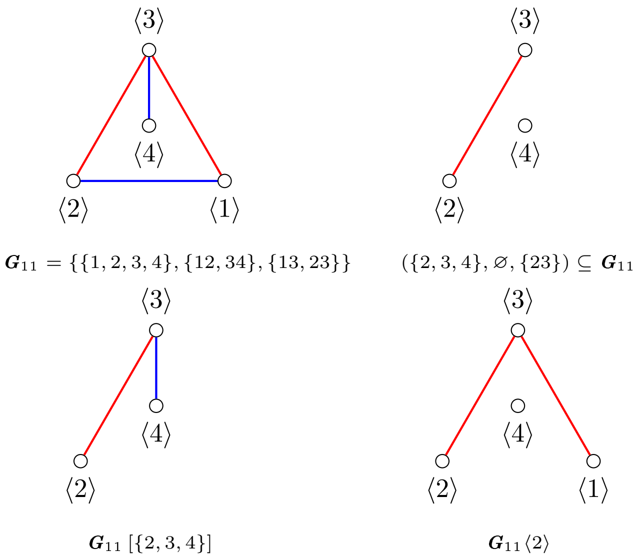
(1) if $p_1 p_2 \leq \frac{1}{cn \log n}$ *or* $\min\{p_1, p_2\} \leq \frac{\log n}{cn}$ *then whp* (G_1, G_2) *does not percolate;*

(2) if $p_1 p_2 \geq \frac{c}{n \log n}$ *and* $\min\{p_1, p_2\} \geq \frac{c \log n}{n}$*, then whp* (G_1, G_2) *percolates.*

Remark 11. This is not the original form of Theorem 10 (see [6]), but it is easy to derive this form from the original.

Remark 12. Connectedness is a necessary (but not sufficient, see Figure 12 and Proposition 4) condition for percolation. Therefore, the condition $\min\{p_1, p_2\} \geq \frac{c \log n}{n}$ in (2) is needed (at least up to the constant c) to guarantee connectedness whp (see Theorem 3).

The aim of this thesis is to extend Theorem 10 to an arbitrary but fixed number of colors. We begin by describing the multi-colored jigsaw process.



$\boldsymbol{G}_{11} = \{\{1,2,3,4\},\{12,34\},\{13,23\}\}$ $(\{2,3,4\},\varnothing,\{23\}) \subseteq \boldsymbol{G}_{11}$

$\boldsymbol{G}_{11}\,[\{2,3,4\}]$ $\boldsymbol{G}_{11}\langle 2\rangle$

Figure 11: Example of a 2-fold graph $\boldsymbol{G}_{11}$ and some other examples (see Definitions 13 and 14).

1.4 The Multi-Colored Jigsaw Algorithm.

We begin with the following definitions:

Definition 13. *An r-fold graph is an $(r+1)$-tuple $\mathbf{G} := (V, E_1, ..., E_r)$, where V is the set of vertices and $E_i \subseteq \binom{V}{2}$ for each $i \in [r]$. We will call $1, 2..., r$ the colors of $\mathbf{G}$ and the graph $G_i = (V, E_i)$ will be said to be of color i for every $i \in [r]$.*

r-fold graph

Definition 14. *Let $\mathbf{G} = (V, E_1, ..., E_r)$ be an r-fold graph and $W \subseteq V$ any subset of vertices. Then we define:*

- *An r-fold subgraph of $\mathbf{G}$ is an r-fold graph of the form $(W, E'_1, ..., E'_r)$ where $E'_i \subseteq E_i[W]$ for every $i \in [r]$. We write $(W, E'_1, ..., E'_r) \subseteq \mathbf{G}$ to say that $(W, E'_1, ..., E'_r)$ is an r-fold subgraph of $\mathbf{G}$.*

  r-fold subgraph

- *The r-fold subgraph induced by W in $\mathbf{G}$ is the r-fold subgraph $\mathbf{G}[W]{:=} (W, E_1[W], ..., E_r[W])$.*

  r-fold subgraph induced by W $\mathbf{G}[W]$

- *An i-fold color-induced subgraph of $\mathbf{G}$ is any i-fold graph of the form $\mathbf{G}\langle j_1, ..., j_i\rangle{:=} (V, E_{j_1}, E_{j_2}, ..., E_{j_i})$, where $1 \le j_1 < j_2 < ... < j_i \le r$.*

  i-fold color-induced subgraph $\mathbf{G}\langle j_1, ..., j_i\rangle$

We can see some examples in Figure 11.

ulti-colored jigsaw algorithm is a generalization of the two-colored jigsaw algorithm: we now merge two clusters iff they share edges of all r colors. In

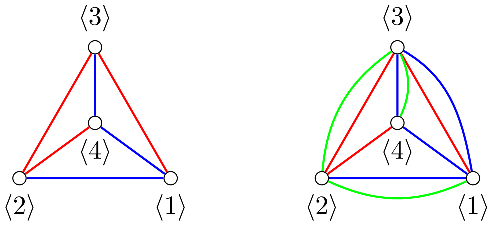


Figure 12: Non-percolating 2 and 3-fold connected 2 and 3-fold graphs. In the case of the 2-fold graph we note that each color graph is connected but there are no clusters sharing at least two edges of different colors, so they cannot be merged. In the case of the 3-fold graph we note that each pair of graphs percolate but the 3-fold graph does not percolate. We need three edges of different colors to merge a pair of clusters, and this condition is clearly not fulfilled for any pair of clusters, so the 3-fold graph cannot percolate.

more detail: let $\boldsymbol{G} = (V, E_1, ..., E_r)$ be an r-fold graph and $G_1, G_2, ..., G_r$ be the colored graphs of $\boldsymbol{G}$. At each discrete time $t = 0, 1, ...$ we have a partition of V into clusters. At time t, let H_t be the graph whose vertices are the clusters, with two vertices joined by an edge iff the corresponding clusters are joined by an edge of each of the r colors. The clusters of our jigsaw process at time $t + 1$ are the unions of the clusters that belong to the same component of H_t. We say that the jigsaw process percolates if we eventually arrive at a single cluster. In particular, for $r = 1$, percolation is equivalent to connectedness (see Proposition 2).

The percolation process is formally described in the following algorithm:

Algorithm 1 (The multi-colored jigsaw algorithm)**.** *For any integer $r \geq 1$. Let $\mathbf{G} := (V, E_1, ..., E_r)$ be an r-fold graph. Jigsaw percolation with input $\mathbf{G}$ evolves at discrete times $t = 0, 1...$ according to the following algorithm. At time t there is a partition $\mathcal{C}_t = \{C_t^1, C_t^2, ..., C_t^{k_t}\}$ of the vertex set V, which we construct inductively as follows:*

1. *We take $k_0 = n$ and $C_0^j := \{j\}$ for all $j \in [n]$. This means that we begin at time 0 with every vertex in a separate set of the partition.*

2. *At time $t \geq 0$, construct a graph H_t on vertex set $\mathcal{C}_t$ by joining $\mathcal{C}_t^i$ to $\mathcal{C}_t^j$ if there exist edges $e_s := \{v_{i,s}, v_{j,s}\} \in E_i$ for all $s \in [r]$ such that $v_{i,s} \in \mathcal{C}_t^i$ and $v_{j,s} \in \mathcal{C}_t^j$.*

3. *If $E(H_t) = \varnothing$, then STOP. Otherwise, construct the partition*

$$\mathcal{C}_{t+1} = \{C_{t+1}^1, ..., C_{t+1}^{k_{t+1}}\},$$

where $C_{t+1}^1, ..., C_{t+1}^{k_{t+1}}$ are the obtained by merging the connected components of H_t i.e. if $D_t^i \subseteq \mathcal{C}_t$ induces a connected component in H_t then $C_{t+1}^i = \bigcup_{C \in D_t^i} C$.

4. If $|\mathcal{C}_{t+1}| = 1$ STOP. Otherwise, go to step 2.

We introduce the following definitions:

Definition 15 (r-fold connected)**.** *We say that an r-fold graph $\mathbf{G} = (V, E_1, ..., E_r)$ is r-fold connected iff the graphs (V, E_i) are connected for each $i \in [r]$.*

r-fold connected

Remark 16. We note that r-fold connectedness does *not* imply percolation when $r \geq 2$, as we can observe in Figure 12. Futhermore, percolation of all i-fold proper color-induced subgraphs (i.e. for all $1 \leq i < r$) does not imply percolation of the the r-fold graph, as we see in the 3-fold graph of Figure 12. On the other hand, percolation *does* imply that every i-fold color-induced subgraph (for $i \in [r]$) percolates (see Propositions 3 and 4).

Percolating r-fold graph

Definition 17.

- *We say that the r-fold graph $\mathbf{G} = (V, E_1, E_2, ..., E_r)$ percolates if Algorithm 1 applied to $\mathbf{G}$ ends with one single cluster. Otherwise we say that $\mathbf{G}$ does not percolate.*

Percolating subset

- *We say that a subset $W \subseteq V$ is a percolating subset (or that it percolates) in $\mathbf{G} = (V, E_1, E_2, ..., E_r)$ if the induced r-fold subgraph $\mathbf{G}[W]$ percolates.*

Remark 18. The definition of a percolating subset corresponds in [6] to the definition of an internally spanned set.

From Algorithm 1 we can prove the following basic properties of percolating r-fold graphs:

Proposition 2. *Percolation for $r = 1$ is equivalent to connectedness.*

Proof. Let G be a 1-fold graph (i.e. a normal graph), then in the first step of Algorithm 1, each connected component of G merges to a cluster, and then the algorithm stops. Therefore, if G is connected we merge all G to a single cluster which means that G percolates. On the other hand, if G percolates we end up with a single cluster that was obtained by merging a connected component, which means that G is connected. □

The following proposition will be used several times in the rest of the thesis:

Proposition 3. *Let $1 \leq i \leq r$. If an r-fold graph $\mathbf{G} = (V, E_1, ..., E_r)$ percolates then every i-fold color-induced subgraph $\mathbf{G}\langle j_1, ..., j_i\rangle$ percolates.*

Proof: Let $\boldsymbol{G}$ be a percolating r-fold graph on n vertices and $\boldsymbol{G}_i$ be any (but fixed) i-fold color-induced subgraph of $\mathbf{G}$.

Recall that Algorithm 1 at time t defines an auxiliary graph H_t on vertices $\mathcal{C}_t$, which are the clusters at time t. In the jigsaw process run on $\boldsymbol{G}$, we call these $\hat{H}_t$ and $\hat{\mathcal{C}}_t$. On the other hand, the corresponding graph and vertices in the jigsaw process run on $\mathbf{G}_i$ will be called $\tilde{H}_i$ and $\tilde{\mathcal{C}}_i$. Let $\hat{t}$ and $\tilde{t}$ be the stopping times of the jigsaw processes of $\mathbf{G}$ and $\mathbf{G}_i$ respectively. Then $\hat{\mathcal{C}}_{\hat{t}} = \{[n]\}$ since $\boldsymbol{G}$ percolates; we aim to prove $\mathbf{G}_i$ also percolates.

In order to avoid some tiresome technical details we slightly redefine Algorithm 1 by letting Step 4 just be "go to step 2". The result is that rather than stopping once no more clusters can be merged, the algorithm keeps outputting the same partition into clusters and (empty) auxiliary graph. This means that the algorithm never terminates, but ensures that H_t and $\mathcal{C}_t$ are defined for all t where $H_t = \mathcal{G}_{t_{\text{end}}}$ and $\mathcal{C}_t = \mathcal{C}_{t_{\text{end}}}$ for any t beyond the stopping time t_{end} of the original algorithm. It is easy to check that an r-fold graph on vertex set V percolates iff $\mathcal{C}_t = \{V\}$ for some sufficiently large t in the modified version of Algorithm 1 (certainly $t = |V|$ is large enough).

Note that $\hat{\mathcal{C}}_t$ and $\tilde{\mathcal{C}}_t$ are partitions of $[n]$ for every t. We prove that for every t, the cluster collection $\hat{\mathcal{C}}_t = \{\hat{C}_t^1, ..., \hat{C}_t^{k_t}\}$ is a refinement of $\tilde{\mathcal{C}}_t = \{\tilde{C}_t^1, ..., \tilde{C}_t^{l_t}\}$ i.e. for every $\hat{C}_t^i \in \hat{\mathcal{C}}_t$ there exists $\tilde{C}_t^{j_i} \in \tilde{\mathcal{C}}_t$ such that $C_t^i \subseteq \tilde{C}_t^{j_i}$. We prove this by induction on t. For $t = 0$ we have by definition (see step 1 of Algorithm 1) that $\hat{\mathcal{C}}_0 = \tilde{\mathcal{C}}_0 = \{\{1\}, ..., \{n\}\}$, this proves the base of induction.

We now prove the inductive step $t \to t + 1$ i.e. we assume that $\hat{\mathcal{C}}_t$ is a refinement of $\tilde{\mathcal{C}}_t$ and aim to prove that $\hat{\mathcal{C}}_{t+1}$ is a refinement of $\tilde{\mathcal{C}}_{t+1}$. By definition, $\hat{\mathcal{C}}_t$ is the vertex set of the graph $\hat{H}_t$ and $\hat{\mathcal{C}}_{t+1}$ is constructed by merging the connected components of $\hat{H}_i$. Recall that two clusters C_t^q, C_t^s are connected in H_t if they share edges of each of the r colors in the input r-fold graph. Therefore if $\hat{C}_t^q \subseteq \tilde{C}_t^{j_q}$ and $\hat{C}_t^s \subseteq \tilde{C}_t^{j_s}$, then $\tilde{C}_t^{j_q}, \tilde{C}_t^{j_s}$ will be connected in $\tilde{H}_t$ (if they are different) since $\hat{C}_t^q, \hat{C}_t^s$ share edges of each of the i colors in the color-induced i-fold subgraph $\boldsymbol{G}_i$ and therefore so do $\tilde{C}_t^{j_q}$ and $\tilde{C}_t^{j_s}$. This means that if for some $W \subseteq [k_t]$, the induced subgraph $\hat{H}_t \left[\bigcup_{w \in W} \{\hat{C}_t^w\}\right]$ is connected, then the induced subgraph $\tilde{H}_t \left[\bigcup_{w \in W} \{\tilde{C}^{j_w}\}\right]$ where $\hat{C}_t^w \subseteq \tilde{C}_t^{j_w}$ will also be connected. In particular, this implies that for every connected connected component $\hat{D}_t^i$ of $\hat{H}_t$ there exists a connected component $\tilde{D}_t^{j_i}$ of $\tilde{H}_t$ such that $\bigcup_{\hat{C} \in \hat{D}_t^i} \hat{C} \subseteq \bigcup_{\tilde{C} \in \tilde{D}_t^{j_i}} \tilde{C}$. This means that $\hat{\mathcal{C}}_{t+1}$ is a refinement of $\tilde{\mathcal{C}}_{t+1}$, proving induction step.

Now since $\hat{\mathcal{C}}_{\hat{t}}, \tilde{\mathcal{C}}_{\hat{t}}$ are partitions of $[n]$ and $\hat{\mathcal{C}}_{\hat{t}} = \{[n]\}$ is a refinement of $\tilde{\mathcal{C}}_{\hat{t}}$, then $\tilde{\mathcal{C}}_{\hat{t}} = \{[n]\}$. This means that $\mathbf{G}_i$ percolates. □

Using Propositions 2 and 3, we can prove that:

Proposition 4. *Every r-fold percolating graph is r-fold connected.*

Proof. Let $\boldsymbol{G} = (V, E_1, ..., E_r)$ be a percolating r-fold graph. By Proposition 3 it follows that the 1-fold subgraphs (V, E_i) are percolating for every $i \in [r]$. Furthermore percolation in the case $r = 1$ is equivalent to connectedness (see Proposition 2). This means that every (V, E_i) is connected and therefore $\boldsymbol{G}$ is r-fold connected. □

1.5 Main Theorem

The following generalization of Theorem 10 is the main result of this thesis:

Theorem 19. *Let $r \in \mathbb{N}$. There exists a constant C_r such that the following holds: suppose that $p_1, ..., p_r$ are functions of n such that $0 \leq p_1 \leq p_2 \leq ... \leq p_r \leq 1$ and $\mathbf{G} = ([n], E_1, E_2, ..., E_r)$ an r-fold graph where $([n], E_i) = G(n, p_i)$ are independent Erdős-Rényi random graphs. For $i \in [r]$ let $P_i := p_1 p_2 ... p_i$.* P_i

Then:

- *(i) If* $P_i \leq \frac{1}{C_r n \log^{i-1} n}$ *for some* $2 \leq i \leq r$ *or* $P_1 \leq \frac{\log n}{C_r n}$ *then whp* $\mathbf{G}$ *does* not *percolate.*
- *(ii) If* $P_i \geq \frac{C_r}{n \log^{i-1} n}$ *for every* $2 \leq i \leq r$ *and* $P_1 \geq \frac{C_r \log n}{n}$*, then whp* $\mathbf{G}$ *percolates.*

We use the following standard definition to differentiate cases (i) and (ii) of Theorem 19:

Definition 20. *If conditions of Theorem 19* (i) *are satisfied, we say that we are in the subcritical case. Similarly, if conditions of Theorem 19* (ii) *are satisfied, we say that we are in the supercritical case.*

subcritical case

supercritical case

We name the graphs that are used in the main theorem as follows:

Definition 21. *An r-fold random graph* $\mathbf{G}(p_1, ..., p_r)$ *is an r-fold graph* $([n], E_1, ..., E_n)$ *where* $([n], E_i) = G(n, p_i)$ *are independent binomial random graphs for every* $i \in [r]$*.*

r-fold random graph

Remark 22. We have seen that if the r-fold graph $\boldsymbol{G}$ percolates then $G_i := (V, E_i)$ must be connected for every $i \in [r]$. From Theorem 3 we know that the threshold for connectivity in random graphs is $\frac{\log n}{n}$. We will ensure the connectivity on each G_i by taking $p_1 = P_1 \geq \frac{C_r \log n}{n}$. More generally, for every $2 \leq i \leq r$ the inequalities $P_i \geq \frac{C_r}{n \log^{i-1} n}$ together with $p_1 \leq p_2 \leq ... \leq p_r$ ensure that every i-fold color-induced subgraph percolate whp. This is a necessary condition for percolation as we saw in Proposition 3.

Remark 23. Since percolation for $r = 1$ is equivalent to connectedness (see Proposition 2), it is clear that Theorem 19 for the case $r = 1$ follows directly from the connectivity threshold (see Theorem 3) by taking any constant $c > 1$. The proofs we are going to present in Sections 2 and 3 are for $r \geq 2$, but could be rewritten to work also in the case $r = 1$. However the proof is far from optimal and the result already very well-known, therefore we will not include it in this thesis.

1.6 Intuition.

We now take a closer look at the jigsaw process for $r = 2$. Suppose that we have a 2-fold random graph $\boldsymbol{G}(n, p_1, p_2)$, where $p_1 p_2 = \frac{c}{n \log n}$ and $\frac{c \log n}{n} \leq p_1 \leq p_2 \leq 1$. At the beginning of the jigsaw process, the clusters consist of single vertices and in order to merge two clusters we need a double edge between them but the average number of double edges incident to a vertex is small $\sim np_1p_2 = \frac{c}{\log n} \ll 1$. This tells us that at the beginning of the jigsaw process, merging clusters can be difficult. However, the crucial point here is that in order to merge clusters, we need double edges between clusters and not between single vertices.

Slightly more precisely, given two clusters C_1, C_2 of sizes k_1 and k_2 respectively, the probability that there are vertices $v_1 \in C_1$ and $v_2 \in C_2$ such that there is a double edge between v_1 and v_2 is approximately:

$$k_1 k_2 p_1 p_2.$$

On the other hand, the probability of having a double edge between the clusters (not between two specific vertices of them) is:

$$\left(1-(1-p_1)^{k_1k_2}\right)\left(1-(1-p_2)^{k_1k_2}\right) \sim (k_1k_2p_1)(k_1k_2p_2) = k_1^2k_2^2p_1p_2,$$

where for the estimate we use the equality:

$$1-(1-p)^k = 1-\left(1-kp+\sum_{i=2}^{k}\binom{k}{i}(-p)^k\right) = kp(1+o(1))$$

valid for $kp = o(1)$. This shows that a big cluster merges is more likely to merge with other clusters than a small cluster. Thus the process may "snowball" in the sense that once two clusters C_1, C_2 merge to form $C_1 \cup C_2$, this new cluster can merge with other clusters that C_1 and C_2 would not have merge individually and so on.

Let us consider the number of clusters $|\mathcal{C}_t|$ that we have at time t in the jigsaw process run in the r-fold graph $\boldsymbol{G}(n, p_1, p_2)$. At time $t = 0$ we have a large number of clusters $|\mathcal{C}_t| = n$, therefore even though the probability of a double edge any two is small, the large number of clusters compensates this fact and allows some clusters to merge. In the subsequent steps the number of clusters that merge decreases. On the other hand, the sizes of the largest clusters increase and we saw that bigger clusters have a better chance of merging. The process can still die out, but beyond a critical size the chances of success i.e. of the cluster eventually merging with all others, giving percolation, are high. In Sections 3.2 and 3.3 we show that if just one cluster gets past a certain size, then the jigsaw process has a high chance of succeeding i.e. the r-fold graph percolates. This phenomenon is known as the "*bottleneck for percolation*".

bottleneck for percolation

1.7 Notation and Observations.

We will ignore floors and ceilings throughout the thesis whenever they do not significantly affect the arguments (this is usually the case since we consider graphs on n vertices, where $n \to \infty$).

The following proposition will be used throughout the thesis for the supercritical case:

Proposition 5. *Let $r, C_r, p_1, p_2, ..., p_r$ satisfy the conditions of Theorem 19 (ii). Then for n large enough there exist real numbers $0 \leq p_1' \leq p_2' \leq ... \leq p_r' \leq 1$ that also satisfy conditions of Theorem 19 (ii) and such that:*

- *$p_i' \leq p_i$ for every i,*

- $p'_1 p'_2 ... p'_r = \frac{C_r}{n \log^{r-1} n}$.

***Proof*.** Let $K_j = \frac{C_r}{n \log^{j-1} n}$ for $2 \leq i \leq r$ and $K_1 = \frac{C_r \log n}{n}$. Consider the following set:

$$\begin{aligned} F = \{(p'_1, ..., p'_r) \in \mathbb{R}^r : \; & K_1 \leq p'_1 \leq p'_2 \leq ... \leq p'_r \leq 1, \\ & P'_j = p'_1 p'_2 ... p'_j \geq K_j \text{ for every } j \in [r], \\ & p'_j \leq p_j \text{ for every } j \in [r]\}, \end{aligned}$$

we note that F is closed because all the inequalities we use to define it are non-strict and it is non-empty because $(p_1, ..., p_r) \in F$. It is also bounded because it is a subset of $[K_1, 1]^r$, therefore F is a non-empty compact set. Thus there is $\vec{p}' := (p'_1, p'_2, ..., p'_r) \in F$ such that $P'_r = p'_1 p'_2 ... p'_r \geq K_r$ is minimal (because P'_r is a continuous funtion of $p'_1, p'_2, ..., p'_r$).

If $P'_r = K_r$, we are done, therefore we will assume that $P'_r > K_r$ and aim to prove a contradiction. If $p'_r > p'_{r-1}$ we could find $p'_{r-1} \leq p''_r < p'_r$ such that $K_r \leq p'_1 p'_2 ... p'_{r-1} p''_r < p'_1 p'_2 ... p'_r$ and thus $(p'_1, p'_2, ..., p'_{r-1}, p''_r) \in F$ (note that no other P'_j is affected). This contradicts the choice of $\vec{p}'$, and therefore we must have $p'_r = p'_{r-1}$.

Let $1 \leq i < r$ be the smallest index such that $p'_r = p'_{r-1} = ... = p'_i$. If $P'_j = p'_1 p'_2 ... p'_j > K_j$ for every $i \leq j \leq r$ we can find $\delta > 0$ such that $(p'_1, p'_2, ..., p'_{i-1}, p'_i - \delta, p'_{i+1}, ..., p'_r) \in F$. This is again contradicts the choice of $\vec{p}'$, therefore an equality $P'_j = K_j$ must hold for some index $i \leq j < r$. Let j be such an index. We note that $j \geq 2$, otherwise $p'_1 = p'_2 = ... = p'_r = K_1$. This is not possible since in this case we would have:

$$P'_r = (K_1)^r = \left(\frac{C_r \log n}{n} \right)^r < \frac{C_r}{n \log^{r-1} n} = K_r$$

for $r \geq 2$ and n large enough. Thus $j \geq 2$.

We recall that $P'_r > K_r$ and $P'_j = K_j$, therefore:

$$\frac{P'_r}{P'_j} > \frac{K_r}{K_j} = \frac{1}{\log^{r-j} n}.$$

On the other hand $\frac{P'_r}{P'_j} = p'_{j+1} p'_{j+2} ... p'_r$, therefore $p'_j = p'_{j+1} = ... = p'_r > \left(\frac{1}{\log^{r-j} n} \right)^{\frac{1}{r-j}} = \frac{1}{\log n}$. Making a case distinction we get:

$$K_{j-1} \leq P'_{j-1} = \frac{P'_j}{p'_j} < \frac{C_r}{n \log^{j-1} n} \log n = \begin{cases} K_{j-1} & \text{if } j \geq 3 \\ \frac{K_1}{\log n} < K_1 & \text{if } j = 2 \end{cases}$$

In either case we get a contradiction. We conclude that $P'_r = K_r$. □

We now prove two very useful propositions that will be used throughout the thesis:

Proposition 6. *Let $\mathcal{P}$ be a monotone increasing property of graphs. If $G(n,p_1), G(n,p_2)$ are two independent binomial random graphs and $p_1 \leq p_2$ then:*

$$\mathbb{P}\left[G(n,p_1) \in \mathcal{P}\right] \leq \mathbb{P}\left[G(n,p_2) \in \mathcal{P}\right].$$

Proof. The proposition is clearly true if $p_1 = 1$, therefore we assume $p_1 < 1$. Given two independent binomial random graphs $G(n,\gamma_1), G(n,\gamma_2)$, we note that $G([n],\gamma_1) \cup G(n,\gamma_2) \sim G(n,\gamma)$ where $\gamma = \gamma_1 + \gamma_2 - \gamma_1\gamma_2$. This is because the edges of the union are independent of each other and an edge exists in $G(n,\gamma_1) \cup G(n,\gamma_2)$ iff exists in at least one $G(n,\gamma_j)$ for $j \in [2]$. From this and the inclusion-exclusion principle we deduce that the union $G(n,\gamma_1) \cup G(n,\gamma_2)$ is a random binomial graph $G(n,\gamma)$ where $\gamma = \gamma_1 + \gamma_2 - \gamma_1\gamma_2$. Thus, if we take an independent binomial random graph $G(n,p^*)$ where p^* is defined by the equation $p_2 = p_1 + p^* - p_1 p^*$ (i.e. $p^* = \frac{p_2 - p_1}{1 - p_1}$), we get $G(n,p_2) \sim G(n,p_1) \cup G(n,p^*)$. Therefore:

$$\mathbb{P}\left[G(n,p_2) \in \mathcal{P}\right] = \mathbb{P}\left[G(n,p_1) \cup G(n,p^*) \in \mathcal{P}\right] \geq \mathbb{P}\left[G(n,p_1) \in \mathcal{P}\right],$$

where the inequality is valid because $\mathcal{P}$ is monotone increasing property. □

Definition 24. *(Union of r-fold graphs) Let $\mathbf{G}_1 = ([n], E_1, E_2, ..., E_r), \mathbf{G}_2 = ([n], E'_1, E'_2, ..., E'_r)$ be r-fold graphs. Then $\mathbf{G}_1 \cup \mathbf{G}_2$ is defined as the r-fold graph* $\mathbf{G}_1 \cup \mathbf{G}_2$
$([n], E_1 \cup E'_1, E_2 \cup E'_2, ..., E_r \cup E'_r)$

Proposition 7. *Let $\mathcal{P}$ be a monotone increasing property of r-fold graphs. If $0 \leq p'_i \leq p_i \leq 1$ for $i \in [r]$, then:*

$$\mathbb{P}\left[\mathbf{G}(n,p'_1,...,p'_r) \in \mathcal{P}\right] \leq \mathbb{P}\left[\mathbf{G}(n,p_1,...,p_r) \in \mathcal{P}\right].$$

Proof. Using the same argument as in Proposition 6 we deduce that for every $i \in [r]$ there is a $0 \leq p^*_i \leq 1$ such that $G([n],p_i) \sim G([n],p'_i) \cup G([n],p^*_i)$. Therefore $\boldsymbol{G}(n,p_1,...,p_r) \sim \boldsymbol{G}(n,p'_1,p'_2,...,p'_r) \cup \mathbf{G}(n,p^*_1,p^*_2,...,p^*_r)$. Thus:

$$\begin{aligned}
&\mathbb{P}\left[\boldsymbol{G}(n,p_1,...,p_r) \in \mathcal{P}\right] \\
&= \mathbb{P}\left[\boldsymbol{G}(n,p'_1,p'_2,...,p'_r) \cup \boldsymbol{G}(n,p^*_1,p^*_2,...,p^*_r) \in \mathcal{P}\right] \\
&\geq \mathbb{P}\left[\boldsymbol{G}(n,p'_1,p'_2,...,p'_r) \in \mathcal{P}\right],
\end{aligned}$$

where the inequality is valid because $\mathcal{P}$ is a monotone increasing property. This completes the proof of Proposition 7. □

Observation 1.

Since percolation is a monotone property, it follows by Proposition 7 that if we prove Theorem 19 for $p'_1, p'_2, ..., p'_r$ such that $p'_i \leq p_i \leq 1$ for every $i \in [r]$ then it will also be valid for $p_1, p_2, ..., p_r$. Therefore by Proposition 6, in the proof of Theorem 19 (ii) we may assume that $P_r = p_1 ... p_r = \frac{C_r}{n \log^{r-1} n}$.

Observation 2.

We can deduce from Observation 1 that $p_2 \leq \left(\frac{p_1 p_2 p_3 ... p_r}{p_1}\right)^{1/(r-1)} \leq \left(\frac{1}{\log n}\right)^{\frac{r}{r-1}}$.

1.8 Overview.

The proof of Theorem 19 is laid out as follows. In Section 2 we prove the subcritical case (part (i)) of Theorem 19 with a simple generalization of the argument used in [6]. This will be a short proof.

In Section 3 we prove the supercritical case (part (ii)) of Theorem 19, which is the main body of the thesis. We will use a similar proof scheme to the one used in [6]. This proof scheme is based on the use of algorithms that construct percolating subsets of increasing size, and is divided into three parts.

In Part I (Section 3.1) we introduce Algorithm 2 which is an algorithm that attempts to construct a percolating subset of size $(\log n)^{1+\varepsilon_r}$ and prove that whp the algorithm succeeds. Part I is the longest and empirically most important in the proof of the supercritical case since we are crossing the bottleneck for percolation (see Section 1.6).

In Part II (Section 3.2) we introduce Algorithm 3 which begins with a percolating subset of size $(\log n)^{1+\varepsilon_r}$ (given by the Part I) and from there attempts to construct a percolating subset of size $\Theta(n)$. Our task is to prove that whp Algorithm 3 succeeds.

In Part III (Section 3.3) of the proof we use the percolating subset of size $\Theta(n)$ given by the second part and the condition $p_1 \geq \frac{C_r \log n}{n}$ to prove that whp the whole set of vertices V percolates.

In Section 4 we will discuss some open problems.

1.9 Main Contributions.

We now summarize the main differences between the two-color case and the multi-colored case:

1. **Upper bounds for $\boldsymbol{p_i}$.** In the two-color case it is easy to get the bound $p_2 \leq (1/\log n)^2$ while in general we can only get $p_2 \leq (1/\log n)^{r/(r-1)}$ (see Observation 2). More generally for $3 \leq i \leq r$ the best possible general upper bound is $p_i \leq 1/\log n$. If we had $p_i \leq (1/\log n)^{1+\delta}$ for every i and for some constant $\delta > 0$, then the calculations would be significantly easier (this is related again to the bottleneck for percolation), but we do not have this in general. Therefore we introduce the parameter i_t (see Definition 32) to cope with this problem. Informally, this new parameter keeps track of the p_i's that are "small", which allows us to make the proofs more compact and clear.

2. **Getting past bottleneck.** The other important difference between the two-color case and the multicolored case is that in Part I of the two-colored case in [6], the size of the percolating subset that the algorithm tries to construct is $(\log n)^{3/2}$. However, the exponent $3/2$ will not work for larger r. We will in fact need an exponent of $1 + \varepsilon_r$, where $\varepsilon_r \to 0$ as $r \to \infty$.

There will be more minor differences between the two-color case and the multi-colored case which will be noted as they appear in the text.

2 Proof of the subcritical case.

In this section we prove part (i) of Theorem 19. We first handle the case when $p_1 \leq \frac{\log n}{C_r n}$. In this case we know from Theorem 3 that the graph (V, E_1) is going to be disconnected whp (taking $C_r > 1$) and from Proposition 4 we know that r-fold connectedness is a necessary condition for percolation. This means that whp the r- fold graph will not percolate, as claimed.

In the case when $P_i \leq \frac{1}{C_r \log^{i-1} n}$ for some $2 \geq i \leq r$, by Proposition 3 we just need to prove that whp the i-fold graph $(V, E_1, ..., E_i)$ does not percolate. To simplify the notation we will replace i by r in the rest of the proof of the subcritical case (since r is arbitrary).

Let $P_r \leq \frac{1}{C_r n \log^{r-1} n}$. We can informally summarize the proof of this section as follows. Given the following events:

$\mathcal{A}$: The r-fold graph $\boldsymbol{G} = (V, E_1, E_2, ..., E_r)$ percolates,

$\mathcal{W}$: There is a subset $W \subseteq V$ of size $\log n \leq |W| \leq 2\log n$, such that $\boldsymbol{G}[W]$ is r-fold connected (see Definitions 15, 17),

we first prove that $\mathcal{A} \subseteq \mathcal{W}$. The second part of the argument is to prove that $\mathbb{P}[\mathcal{W}] = o(1)$ given that $P_r = p_1 p_2 ... p_r$ is small. Since $\mathbb{P}[\mathcal{A}] \leq \mathbb{P}[\mathcal{W}]$ we can conclude that the event $\mathcal{A}$ is unlikely.

We begin by proving that the event $\mathcal{W}$ contains the event $\mathcal{A}$.

Claim 25. *For every natural number $k \leq n/2$ and every percolating r-fold graph $\mathbf{G} = (V, E_1, E_2, ..., E_r)$ there is a subset W of V of size at least k but not larger than $2k$ such that $\mathbf{G}[W]$ is r-fold connected.*

Proof. We can split the percolation process defined in Algorithm 1 into smaller steps, in each of which two clusters (i.e. elements of the current partition $\mathcal{C}_t$) merge. Specifically, we can modify step (3) in Algorithm 1 so that instead of merging entire connected components of sets C_t^i (connected in $\mathcal{G}_t$), we only merge an arbitrary pair of sets C_t^i which are joined by an edge of each of the r colors.

At the beginning of the percolation process, each vertex of V is a cluster by itself and since the r-fold graph percolates, the process ends with a cluster of size $n \geq k$. We therefore begin to merge pairs of clusters and consider the first time that a cluster W of size at least k appears. This cluster cannot be of size larger than $2k$ since it is the union of two clusters of size less than k.

Furthermore, since $\boldsymbol{G}[W]$ is percolating we know by Proposition 4 that it is r-fold connected. This completes the proof of Claim 25. □

Let $\mathcal{W}_k$ be the event that there exists a subset W of V such that $|W| = k$ and $\boldsymbol{G}[W]$ is r-fold connected.

Let $\mathcal{W} := \bigcup_{k=\log n}^{2\log n} \mathcal{W}_k$, then by Claim 25 and the union bound:

$$\mathbb{P}[\mathcal{A}] \le \mathbb{P}[\mathcal{W}] \le \sum_{k=\log n}^{2\log n} \mathbb{P}[\mathcal{W}_k]. \tag{1}$$

We define the random variable $\mathcal{T}_k$ to be the number of r-fold subgraphs $(W, E_1', E_2', ..., E_r')$ where $W \subseteq V$ and $E_i' \subseteq E_i$ and the graphs (W, E_i') are trees with exactly k vertices. We can deduce using Markov's inequality (which states that $\alpha\mathbb{P}[\mathcal{X} \ge \alpha] \le \mathbb{E}[X]$ for a non-negative random variable, see e.g. [1]) on $\mathcal{T}_k$ that:

$$\mathbb{P}[\mathcal{W}_k] = \mathbb{P}[\mathcal{T}_k \ge 1] \le \mathbb{E}[\mathcal{T}_k]. \tag{2}$$

We now want to estimate $\mathbb{E}[\mathcal{T}_k]$. We begin by calculating the number T of possible $(r+1)$-tuples $(W, D_1, D_2, ..., D_r)$ where $W \subseteq V$ is a subset of vertices, $D_i \subseteq \binom{W}{2}$ is any subset of edges and the graphs (W, D_i) form trees with exactly k vertices. We recall that there are k^{k-2} labeled spanning trees in a complete graph with k vertices. Therefore, for a given W there are k^{k-2} possible choices for each D_i. This holds for each possible subset W with exactly k vertices of V, of which there are $\binom{n}{k}$, therefore $T = \binom{n}{k}\prod_{i=1}^{r} k^{k-2}$. Recalling that the graphs (V, E_i) are independent of each other and each of these $(r+1)$-tuples $(W, D_1, ..., D_r)$ have $k-1$ edges for each color, we deduce that the probability that these subgraphs exist in $\boldsymbol{G}$ is $\prod_{i=1}^{r} p_i^{k-1}$. Thus $\mathbb{E}[\mathcal{T}_k] = \binom{n}{k}\prod_{i=1}^{r} k^{k-2}p_i^{k-1}$.

Using the previous estimate for $\mathbb{E}[\mathcal{T}_k]$ and the well-known bound (see e.g. [8]) $\binom{n}{k} \le (en/k)^k$ we conclude that:

$$\begin{aligned}
\mathbb{P}[\mathcal{A}] \overset{(1)}{\le} \sum_{k=\log n}^{2\log n} \mathbb{P}[\mathcal{W}_k] \overset{(2)}{\le} \sum_{k=\log n}^{2\log n} \mathbb{E}[\mathcal{T}_k]& \\
&= \sum_{k=\log n}^{2\log n} \binom{n}{k} \prod_{i=1}^{r} k^{k-2} p_i^{k-1} \\
&\le \frac{1}{P_r} \sum_{k=\log n}^{2\log n} \left(\frac{en}{k}\right)^k k^{r(k-2)} P_r^k \\
&\le \frac{1}{P_r} \sum_{k=\log n}^{2\log n} (enk^{r-1}P_r)^k \\
&\le \frac{1}{P_r} \sum_{k=\log n}^{2\log n} (2^{r-1}en(\log n)^{r-1}P_r)^k \\
&= 2^{r-1}en(\log n)^{r-1} \sum_{k=\log n}^{2\log n} (2^{r-1}en(\log n)^{r-1}P_r)^{k-1}.
\end{aligned}$$

We recall that $P_r \leq 1/(C_r n \log^{r-1} n)$. Assuming that $C_r \geq 2^{r-1}e^3$ we get $2^{r-1}en(\log n)^{r-1}P_r \leq 1/e^2$. We deduce that:

$$\begin{aligned}
\mathbb{P}[\mathcal{A}] \leq 2^{r-1}en(\log n)^{r-1} \sum_{k=\log n}^{2\log n} \left(\frac{1}{e^2}\right)^{k-1} &\leq n^{3/2} \sum_{k=\log n}^{\infty} \left(\frac{1}{e^2}\right)^{k-1} \\
&= n^{3/2} \left(\frac{1}{e^2}\right)^{\log n - 1} \frac{1}{1-\frac{1}{e^2}} \\
&\leq \frac{e^4 n^{3/2}}{n^2(e^2-1)} \\
&= o(1).
\end{aligned}$$

This completes the proof of the subcritical case. □

Remark 26. It is worth noting that we for the proof we exploit the bottleneck for percolation by disproving the existence of a percolating subset of size $\sim \log n$. The choice of $\log n$ was sensible precisely because this is where the bottleneck occurs.

3 Proof of the supercritical case.

In this section we will prove part (ii) of Theorem 19.

The main idea for the proof is to construct an increasing sequence of percolating subsets $V_1 \subseteq V_2 \subseteq V_3 = V$, where V_1 and V_2 are themselves constructed by increasing sequences of percolating subsets. Therefore we will divide the proof into three parts:

> Part I: We show that whp there is a percolating subset $V_1 \subseteq V$ of size at least $t_1 := (\log n)^{1+\varepsilon_r}$ for some fixed $\varepsilon_r > 0$, dependent only on the number of colors which will be defined later. t_1
>
> Part II: In this part we show that conditioned on the existence of a percolating subset $V_1 \subseteq V$ of size at least t_1, whp there exists a percolating subset $V_2 \supset V_1$ of size at least $n/2^{r+2}$.
>
> Part III: Finally we show that conditioned on the existence of a percolating subset V_2 of size at least $n/2^{r+2}$, whp the whole set V percolates.

We prove Parts I and II by defining exploration algorithms that will expose the edges of the r-fold graph in a special order that depends on what has been observed so far in the process. The independence between the three parts of the proof is guaranteed by independent rounds of exposure.

Let $\boldsymbol{G} = \boldsymbol{G}(n, p_1, ..., p_r)$ an r-fold graph that satisfies the conditions of Theorem 19. From Proposition 5, we may assume $P_r = p_1 \cdot p_2 \cdot ... \cdot p_r = \frac{C_r}{n \log^{r-1} n}$.

For $i = 1, 2, ..., r$ and $j = 1, 2, 3$ let:

- G_i^j be a binomial random graph $G(n, p_i^*)$ where $p_i^* := \frac{p_i}{3}$ (independently for each i, j),
- $G_i := G_i^1 \cup G_i^2 \cup G_i^3$,
- $\boldsymbol{G}^{(j)} := ([n], E_1^{(j)}, E_2^{(j)}, ..., E_r^{(j)})$ where $E_i^{(j)} := E(G_i^{(j)})$.

In Part j of the proof we will work with the round of exposure $\boldsymbol{G}^{(j)}$. Therefore all three parts of the proof will be independent.

By proving that whp all three parts of the proof hold for their respective round of exposure $\boldsymbol{G}^{(j)}$, we are proving that the r-fold graph $\boldsymbol{G}^* = \boldsymbol{G}^{(1)} \cup \boldsymbol{G}^{(2)} \cup \boldsymbol{G}^{(3)}$ percolates whp. Using a similar argument as in Proposition 6, we note that $G_i \sim G(n, p_i')$ where $p_i' := 3p_i^* - 3(p_i^*)^2 + (p_i^*)^3 = p_i^*(3 - 3p_i^* + (p_i^*)^2)$, therefore:

- $\boldsymbol{G}^* \sim \boldsymbol{G}' := \boldsymbol{G}(n, p_1', p_2', ..., p_r')$,
- $p_i' \leq 3p_i^* = p_i$ for every $i \in [r]$.

From this and Proposition 7, it follows that:

$$\mathbb{P}\left[\boldsymbol{G}^* \text{ percolates}\right] = \mathbb{P}\left[\boldsymbol{G}' \text{ percolates}\right] \leq \mathbb{P}\left[\boldsymbol{G} \text{ percolates}\right],$$

thus if $\boldsymbol{G}^*$ percolates whp, then $\boldsymbol{G}$ will also percolate whp.

Observation 3 (Notation)**.** *Renaming the constant $C_r := 3^r c_r^{r-1}$, we note that:*

- $P_j^* = p_1^* p_2^* ... p_j^* = \frac{p_1 p_2 ... p_j}{3^j} \geq \frac{c_r^{r-1}}{n \log^{j-1} n}$,
- $P_1^* = p_1^* = \frac{p_1}{3} \geq \frac{c_r^{r-1} \log n}{n}$,
- $P_r^* = p_j^* p_j^* ... p_r^* = \frac{p_1 p_2 ... p_r}{3^r} = \frac{c_r^{r-1}}{n \log^{j-1} n}$,

which means that $p_1^*, p_2^*, ..., p_r^*$ satisfy the conditions of Theorem 19 with the constant c_r^{r-1} and also the equality $P_r^* = \frac{c_r^{r-1}}{n \log^{r-1} n}$. Throughout the rest of the thesis we will work with $p_1^*, p_2^*, ..., p_r^*$ and the constant c_r^{r-1}, but ***in order to simplify of notation we will use $\mathbf{p_i}$ to denote $\mathbf{p_i^*}$ and $\mathbf{C_r}$ to denote $\mathbf{c_r^{r-1}}$.***

Note: *this is abuse of notation as p_i and C_r already have a different definition. However, since we only aim to prove the existence of a constant C_r, and do not aim to determine its value, multiplication by some other constant does not change things in any fundamental way. We can recover the original values with the equalities $\mathbf{p_i = 3p_i^*}$ and $\mathbf{C_r = 3^r c_r^{r-1}}$.*

Remark 27.

- In terms of length of the proofs, Part I is the longest and empirically most important of them all. This is because in Part I we pass the bottleneck for percolation, i.e. we ensure the existence whp of a percolating subset of size slightly larger than $\log n$. Once we pass the bottleneck, Parts II and III will be relatively short and easy, with Part III being the shortest and easiest to prove.

3.1 Part I

We will construct a large percolating subset V_1 by "trial and error". Algorithm 2 will start form a single vertex and add one vertex at a time to attempt to construct V_1. If we only try once, this process has a low probability of success but since we try many times, it has a good chance of succeeding i.e. succeeds whp. We divide the proof into two stages:

- First, we will bound from below the probability that the algorithm constructs a percolating subset of size at least $t_0 := \frac{\log n}{c_r}$ (in one trial, see Lemma 36).
- Second, conditioned on the algorithm constructing a percolating subset of size at least t_0, we will bound from below the probability that the algorithm constructs a percolating subset of size at least $t_1 = (\log n)^{1+\varepsilon_r}$ (in one trial, see Lemma 37).

The probability that Algorithm 2 reaches t_1 in one trial is bounded from below by the product of the probabilities of the previous stages. This product turns out to be small but the crucial point here is that Algorithm 2 makes many attempts to reach t_1. The probability that at least one of these attempts succeeds will tend to 1 as $n \to \infty$, proving that whp Algorithm 2 constructs a percolating subset of size t_1 (see Lemma 39).

We now describe, in an informal way, how Algorithm 2 works. In round k we add one vertex at a time to the trial set (X_k^t), which will always be a percolating set. If the algorithm finds a suitable vertex to add to the trial set (Step 4 below) we proceed to the next t-step. If not, we discard all the vertices of the trial set and begin a new round.

Here is the formal description of the algorithm we are going to use for the rest of Part I.

Algorithm 2 (The 1-by-1 algorithm)**.** *The algorithm is divided into rounds, indexed by k, and each round is divided into steps, indexed by t. At the start of the k-th round there is a set $A_k^0 \subseteq [n]$ of active vertices and a set $D_k \subseteq [n]$ of discarded vertices. We begin with $A_1^0 = [n]$ and $D_1 = \varnothing$. The procedure of the k-th round is as follows:*

(1) At the start of the t-th step of the k-th round there are sets:

- $X_k^t = \{x_k^1, x_k^2, ..., x_k^t\} \subseteq A_k$ *a set of trial vertices.*
- $U_k^t \subseteq A_k^0$ *a set of dormant vertices.*

Where $A_k^0 = X_k^t \sqcup A_k^t \sqcup U_k^t$.

(2) For $t = 0$, we move an arbitrary active vertex to the trial set:

- $X_k^1 := \{x_k^1\}$.
- $U_k^1 := \varnothing$.
- $A_k^1 := A_k^0 \backslash x_k^1$, *where* $x_k^1 \in A_k^0$ *is arbitrary.*
- $R_k^0 := \varnothing$.

(3) For $t \geq 1$, we reveal all edges of $\mathbf{G}^{(1)}$ between A_k^t and X_k^t and let:

- $R_k^t := \{x \in A_k^t : xx_k^t \in E_1^{(1)}\}$.
- $B_k^t := \{x \in R_k^t :$ *for every* $i \in \{2, 3, ..., r\}$ *there exists* $s_i \leq t$ *such that* $xx_k^{s_i} \in E_i^{(1)}\}$.

(4) If $B_k^t \neq \varnothing$, then let x_k^{t+1} be an arbitrary element of B_k^t. Then set:

- $X_k^{t+1} := X_k^t \cup \{x_k^{t+1}\}$
- $A_k^{t+1} := A_k^t \backslash R_k^t$.
- $U_k^{t+1} := U_k^t \cup \left(R_k^t \backslash \{x_k^{t+1}\}\right)$.

If $t \geq t_1 = (\log n)^{1+\varepsilon_r}$ then STOP, otherwise set $t = t + 1$ and go to step (3).

(5) If $B_k^t = \varnothing$, then set

- $A_{k+1} := A_k \backslash X_k$
- $D_{k+1} := D_k \cup X_k^t$

If

$$k \geq \frac{n}{2(\log n)^{1+\varepsilon_r}}$$

then STOP, otherwise set $k := k+1$ *and* $t := 1$, *and go to step (1).*

It is important to note that since every tested edge has at least one of its endponts in the trial set, we guarantee independence between rounds by dismissing the trial set at the end of each round. We also have independence within each round, because no edge is tested twice within a round. This is because we first construct the set R_k^t by testing the edges of the first color that exists between the last added vertex of the trial set and the active vertices A_k^t. We then proceed to reveal all the edges of the other colors between the trial set and R_k^t. The crucial point is that if $B_k^t \neq \varnothing$, we discard the set R_k^t from the active vertices by storing them in a set of "dormant vertices" U_k^t. This will guarantee that no edge is tested more than once within a round, but the "dormant vertices" will be active again in the next round (if there is one). We note that all the tested edges within this t-step had at least one endpoint in R_k^t and therefore they will be no longer tested in the next t-step. This guarantees independence within a round.

Remark 28. Since we consider at most $n/(2(\log n)^{1+\varepsilon_r})$ rounds, and stop each with a trial set of size at most $(\log n)^{1+\varepsilon_r}$ vertices, we start each new round with at least $n/2$ vertices:

$$|A_k| := |A_k^0| \geq \frac{n}{2}.$$

We will need the following definitions:

Definition 29.
- *Let* $\mathcal{X}_k^t$ *be the event that* X_k^t *is defined (i.e. we reach step* t *in round* k*).*
- *Let* $\mathcal{S}_k^t := \{|R_k^s| \leq \frac{n}{4t_1}$ *for* $s = 0, 1, 2, ..., t\}$.
- *Let* $\mathcal{Y}_k^t := \mathcal{X}_k^t \cap \mathcal{S}_k^t$.
- *Let* $r_k^t := \mathbb{P}\left[\mathcal{Y}_k^t \middle| \mathcal{Y}_k^{t-1}\right]$ *for* $k \leq n/(2(\log n)^{1+\varepsilon_r})$ *and* $t \geq 1$.

Remark 30.
- The event $\mathcal{X}_k^t$ means that we found a percolating subset of size t formed with only edges of the first round of exposure. Conditioned on getting to round k the event $\mathcal{X}_k^1$ always holds. For $t \geq 2$ the event $\mathcal{X}_k^t$ is equivalent to the event that B_k^{t-1} is non-empty.

- The event $\mathcal{S}_k^t$ guarantees that within a round k, we do not discard too many vertices by step t. More specifically, if the event $\mathcal{Y}_k^{t-1}$ holds we have:
$$|A_k^t| \geq |A_k^0| - (t-1)n/4t_1 \geq \frac{n}{2} - \frac{t}{t_1}\frac{n}{4} \geq \frac{n}{4}.$$
We note that if we get to round k the event $\mathcal{S}_k^0$ always holds.

We will use the following technical claim to approximate some expressions:

Claim 31. *For $t \geq 0$, $p \leq 1$ we have:*

a) If $1 - pt \geq 0$ then $1 - (1-p)^t \geq pt(1-pt)$.

b) If $1 - pt \leq \frac{1}{2}$ then $1 - (1-p)^t \geq \frac{1}{5}$.

Proof. **a)** $1 - (1-p)^t \geq 1 - e^{-pt}$
$$\begin{aligned} &= 1 - \left(1 - pt + \frac{(pt)^2}{2!} - \frac{(pt)^3}{3!} + \frac{(pt)^4}{4!} - ...\right) \\ &= 1 - \left(1 - pt + \frac{(pt)^2}{2!} - \sum_{i\geq 1} \frac{(pt)^{2i+1}}{(2i+1)!}\left(1 - \frac{pt}{2i+2}\right)\right) \\ &\geq pt - \frac{(pt)^2}{2} \\ &\geq pt(1-pt), \end{aligned}$$

where we use $(1-p)^t \leq e^{-pt}$ for the first inequality, which is valid since $t \geq 0$. For the second inequality we note that the terms $\left(1 - \frac{pt}{2i+2}\right)$ are non-negative since $pt \leq 1$.

b) If $1 - pt \leq \frac{1}{2}$ then $e^{-pt} \leq e^{-\frac{1}{2}}$ and therefore:
$$1 - (1-p)^t \geq 1 - e^{-pt} \geq 1 - e^{-1/2} > 1/5$$
Where for the first inequality we used $(1-p)^t \leq e^{-pt}$, which is valid since $t \geq 0$. □

We will use the following observation:

Observation 4. *Let $\varepsilon_r := \frac{1}{r} < \frac{1}{r-1}$. Using the bound $p_2 \leq \left(\frac{1}{\log n}\right)^{\frac{r}{r-1}}$ of Observation 2, for $t \leq t_1$ we can deduce that:*
$$p_2 t \leq p_2 t_1 \leq \frac{(\log n)^{1+\varepsilon_r}}{(\log n)^{\frac{r}{r-1}}} = (\log n)^{\varepsilon_r - \frac{1}{r-1}} = o(1).$$

The following parameter will be very useful in what is still left to prove:

Definition 32. *Let* $i_t := \max\{i \in [2, r] : 1 - p_i t \geq \frac{1}{2}\}$.

Remark 33. From Observation 4 we have that $p_2 t_1 = o(1)$. This means that i_t is well defined for $t \leq t_1$ and n large enough.

We now want to calculate a lower bound on the probability of "one-step success" i.e. the probability of being able to add a vertex to the percolating set that is under construction in the t-th step of the k-th round of Algorithm 2. We calculate this lower bound by first calculating a lower bound on the probability that an active vertex is in B_k^t.

Lemma 34. *For n large enough and $1 \leq t \leq t_1 = (\log n)^{1+\varepsilon_r}$ we have that independently for each $x \in \mathcal{A}_k^t$ it holds the following:*

$$\mathbb{P}[x \in B_k^t] \geq \left(\frac{1}{5}\right)^{r-1} P_{i_t} t^{i_t - 1},$$

where $P_{i_t} = p_1 p_2 ... p_{i_t}$ (already defined in Theorem 19).

Proof. We begin by observing the following identity:

$$\mathbb{P}[x \in B_k^t] = p_1 \prod_{j=2}^{r} (1 - (1 - p_j)^t).$$

Using the definition of i_t and Claim 31 a) and b) on the elements of the products $\prod_{j=2}^{i_t}(1 - (1 - p_j)^t)$ and $\prod_{j=i_t+1}^{r}(1 - (1 - p_j)^t)$ respectively we get:

$$\begin{aligned}\mathbb{P}[x \in B_k^t] &\geq p_1 \left(\frac{1}{5}\right)^{r-i_t} \prod_{j=2}^{i_t} p_j t(1 - p_j t) \\ &\geq \left(\frac{1}{2}\right)^{i_t-1} \left(\frac{1}{5}\right)^{r-i_t} P_{i_t} t^{i_t-1} \\ &\geq \left(\frac{1}{5}\right)^{r-1} P_{i_t} t^{i_t-1}.\end{aligned}$$

□

We now make use of independence and the lower bound of Lemma 34.

Lemma 35. *For $1 \leq t \leq t_1 = (\log n)^{1+\varepsilon_r}$ and n large enough the following holds:*

(a) $\mathbb{P}\left[\mathcal{Y}_k^t | \mathcal{Y}_k^{t-1}\right] \geq 1 - \exp\left\{-\left(\frac{1}{6}\right)^r \left(\frac{c_r t}{\log n}\right)^{i_t-1}\right\}$.

(b) If $\left(\frac{1}{6}\right)^r \left(\frac{c_r t}{\log n}\right)^{i_t-1} \leq 1$ *we have:*

$$\mathbb{P}\left[\mathcal{Y}_k^t | \mathcal{Y}_k^{t-1}\right] \geq \left(\frac{1}{12}\right)^r \left(\frac{c_r t}{\log n}\right)^{i_t-1}.$$

***Proof*.** Since case (b) follows from case (a) and the inequality $1-\frac{x}{2} \geq \exp(-x)$, valid for $x \in [0,1]$, we only need to prove case (a).

We recall that $\mathcal{Y}_k^t = \mathcal{X}_k^t \cap \mathcal{S}_k^t$, thus:

$$\mathbb{P}\left[\mathcal{Y}_k^t | \mathcal{Y}_k^{t-1}\right] \geq 1 - \mathbb{P}\left[\overline{\mathcal{X}_k^t} | \mathcal{Y}_k^{t-1}\right] - \mathbb{P}\left[\overline{\mathcal{S}_k^t} | \mathcal{Y}_k^{t-1}\right].$$

We now need to estimate from above the probability terms on the right side of the inequality.

First estimate: Let $\mathcal{Z}_k^t$ be the random variable that represents the number of sets Z of size $\frac{n}{4t_1}$ such that $Z \subseteq R_k^t$. We deduce using Markov's inequality that:

$$\begin{aligned}
\mathbb{P}\left[\overline{\mathcal{S}_k^t} | \mathcal{Y}_k^{t-1}\right] &= \mathbb{P}\left[|R_k^t| > \frac{n}{4t_1} \Big| \mathcal{Y}_k^{t-1}\right] \\
&\leq \mathbb{P}\left[\mathcal{Z}_k^t \geq 1 \Big| \mathcal{Y}_k^{t-1}\right] \\
&\leq \mathbb{E}\left[\mathcal{Z}_k^t \Big| \mathcal{Y}_k^{t-1}\right] \\
&\leq \binom{n}{n/(4t_1)} p_1^{n/(4t_1)} \\
&\leq (4et_1)^{n/(4t_1)} p_1^{n/(4t_1)} \leq (12p_1t_1)^{n/(4t_1)} \leq e^{-\sqrt{n}},
\end{aligned}$$

where for the second inequality, we use the bound $\binom{n}{k} \leq (en/k)^k$. For the last inequality we use that $n/(4t_1) \geq \sqrt{n}$ and $p_1t_1 = o(1) \leq \frac{1}{e}$ for n large enough (see Observation 4). Note that these approximations are rather crude, but will be sufficient since the term with $\overline{\mathcal{X}_k^t}$ will be the more critical one.

Second estimate: Using Lemma 34 and the observation that $|A_k^{t-1}| \geq n/4$ made in Remark 28 we get:

$$\begin{aligned}
\mathbb{P}\left[\overline{\mathcal{X}_k^t} | \mathcal{Y}_k^{t-1}\right] &= \prod_{x \in A_k^{t-1}} \mathbb{P}[x \notin B_k^{t-1}] \\
&\leq \left(1 - \left(\frac{1}{5}\right)^{r-1} P_{i_t} t^{i_t-1}\right)^{\frac{n}{4}} \\
&\leq \exp\left\{-\frac{1}{4}\left(\frac{1}{5}\right)^{r-1} nP_{i_t} t^{i_t-1}\right\}.
\end{aligned}$$

From the assumptions of Theorem 19 and Observation 3, we have that $P_{i_t} \geq C_r/(n \log^{i_t-1} n) \geq c_r^{i_t-1}/(n \log^{i_t-1} n)$. Thus we get:

$$\mathbb{P}\left[\overline{\mathcal{X}_k^t} | \mathcal{Y}_k^{t-1}\right] \leq \exp\left\{-\left(\frac{1}{5}\right)^r \left(\frac{c_r t}{\log n}\right)^{i_t-1}\right\}.$$

Combining both estimates we get:

$$\mathbb{P}\left[\mathcal{Y}_k^t | \mathcal{Y}_k^{t-1}\right] \geq 1 - \exp\left\{-\left(\frac{1}{5}\right)^r \left(\frac{c_r t}{\log n}\right)^{i_t - 1}\right\} - \exp\{-\sqrt{n}\}.$$

To complete the proof we recall that $t \leq t_1$, $2 \leq i_t \leq r$ and observe that:

$$\left(\frac{t}{\log n}\right)^{i_t-1} \leq \left(\frac{t_1}{\log n}\right)^{i_t-1} \leq \left(\frac{t_1}{\log n}\right)^{r-1} = (\log n)^{\varepsilon_r (r-1)} = o(\sqrt{n}),$$

from here we conclude that:

$$\mathbb{P}\left[\mathcal{Y}_k^t | \mathcal{Y}_k^{t-1}\right] \geq 1 - \exp\left\{-\left(\frac{1}{6}\right)^r \left(\frac{c_r t}{\log n}\right)^{i_t - 1}\right\}.$$

This completes the proof of Lemma 35. □

Recall that $t_0 = \frac{\log n}{c_r}$. In order to calculate a lower bound on the probability of "proceeding to step t_1" we calculate lower bounds for the events "proceeding to step t_0" and "proceeding to step t_1 given that we already proceeded to step t_0". We formally express this in Lemmas 36 and 37. The proof of these lemmas is by means of independence and the lower bounds of Lemma 35.

Lemma 36. $\mathbb{P}\left[\mathcal{Y}_k^{t_0} | \mathcal{X}_k^0\right] \geq n^{-6(r-1)/c_r}$.

Proof. We begin by observing the following equalities:

$$\mathbb{P}\left[\mathcal{Y}_k^{t_0} | \mathcal{X}_k^0\right] = \mathbb{P}\left[\mathcal{Y}_k^{t_0} | \mathcal{Y}_k^0\right] = \prod_{t=1}^{t_0} \mathbb{P}\left[\mathcal{Y}_k^t | \mathcal{Y}_k^{t-1}\right].$$

We note that $\frac{c_r t}{\log n} \leq 1$ for $1 \leq t \leq t_0$. Therefore we can use Lemma 35 (b):

$$\begin{aligned} \mathbb{P}\left[\mathcal{Y}_k^{t_0} | \mathcal{X}_k^0\right] &\geq \prod_{t=1}^{t_0} \left(\frac{1}{12}\right)^r \left(\frac{c_r t}{\log n}\right)^{i_t - 1} \\ &\geq \prod_{t=1}^{t_0} \left(\frac{1}{12^2}\right)^{r-1} \left(\frac{c_r t}{\log n}\right)^{r-1} \\ &\geq \left(\frac{c_r}{144 \log n}\right)^{(r-1)t_0} (t_0!)^{r-1}. \end{aligned}$$

Using the well known inequality $t! \geq (t/e)^t$ valid for all $n \in \mathbb{N}$, we get:

$$\begin{aligned} \mathbb{P}\left[\mathcal{Y}_k^{t_0} | \mathcal{X}_k^0\right] &\geq \left(\frac{1}{144e}\right)^{(r-1)\frac{\log n}{c_r}} \\ &\geq n^{-\frac{6(r-1)}{c_r}}, \end{aligned}$$

since $\frac{1}{144e} \geq \frac{1}{e^6}$. This completes the proof of Lemma 36. □

Lemma 37. $\mathbb{P}\left[\mathcal{X}_k^{t_1}|\mathcal{Y}_k^{t_0}\right] \geq n^{-O(1)/c_r}$.

Proof. We begin by applying Lemma 35 a):

$$\begin{aligned}\mathbb{P}\left[\mathcal{X}_k^{t_1}|\mathcal{Y}_k^{t_0}\right] &= \prod_{t=t_0+1}^{t_1} \mathbb{P}\left[\mathcal{Y}_k^{t}|\mathcal{Y}_k^{t-1}\right] \\ &\geq \prod_{t=t_0}^{t_1}\left(1-\exp\left\{-\left(\frac{1}{6}\right)^r\left(\frac{c_r t}{\log n}\right)^{i_t-1}\right\}\right).\end{aligned}$$

Suppose $f(r) \geq 1$. We use the inequality $1-x \geq e^{-f(r)x}$, valid (at least) for $0 \leq x \leq 1-\frac{1}{f(r)}$ (we will show this at the end of the proof). Setting $x = \exp\left\{-\left(\frac{1}{6}\right)^r\left(\frac{c_r t}{\log n}\right)^{i_t-1}\right\}$ and noting that $\frac{c_r t}{\log n} \geq \frac{c_r t_0}{\log n} = 1$ for $t \geq t_0$ we deduce that $x \leq \exp\left\{-\left(\frac{1}{6}\right)^r\right\}$, therefore we can set $f(r) = \frac{1}{1-\exp\left\{-\left(\frac{1}{6}\right)^r\right\}} > 1$. Thus:

$$\begin{aligned}\mathbb{P}\left[\mathcal{X}_k^{t_1}|\mathcal{Y}_k^{t_0}\right] &\geq \exp\left(-f(r)\sum_{t=t_0}^{t_1}\exp\left\{-\left(\frac{1}{6}\right)^r\left(\frac{c_r t}{\log n}\right)^{i_t-1}\right\}\right) \\ &\geq \exp\left(-f(r)\sum_{t=t_0}^{t_1}\exp\left\{-\left(\frac{1}{6}\right)^r\left(\frac{c_r t}{\log n}\right)\right\}\right) \\ &\geq \exp\left(-\frac{f(r)\exp\left\{-\left(\frac{1}{6}\right)^r\right\}}{1-\exp\left\{-\left(\frac{1}{6}\right)^r\frac{c_r}{\log n}\right\}}\right),\end{aligned}$$

where in the last inequality, we use the formula for the (infinite) sum of a geometric series.

We now simplify the denominator by using the inequality $e^{-x} \leq 1-x/2$ valid for $x \leq 1$:

$$\begin{aligned}\mathbb{P}\left[\mathcal{X}_k^{t_1}|\mathcal{Y}_k^{t_0}\right] &\geq \exp\left(-\frac{f(r)\exp\left\{-\left(\frac{1}{6}\right)^r\right\}}{1-\left(1-\frac{1}{2}\left(\frac{1}{6}\right)^r\frac{c_r}{\log n}\right)}\right) \\ &= \exp\left(-\frac{2^{r+1}3^r f(r)\exp\left(-\left(\frac{1}{6}\right)^r\right)\log n}{c_r}\right) = n^{-O(1)/c_r},\end{aligned}$$

where the $O(1)$ term is a constant that depends on r but *not* on c_r. To complete the proof of Lemma 37 it is only left to prove that given $a \geq 1$ the inequality $1-x \geq e^{-ax}$ is valid (at least) for $0 \leq x \leq 1-\frac{1}{a}$. Indeed, let $g(x) := 1-x-e^{-ax}$, then $g''(x) = -a^2e^{-ax} < 0$ which means that g is concave. This means that:

$$g\left(\underbrace{(1-\gamma)\cdot 0}_{=0}+\gamma\cdot\left(1-\frac{1}{a}\right)\right) \geq \underbrace{(1-\gamma)\cdot g(0)}_{=0}+\gamma\cdot g\left(1-\frac{1}{a}\right),$$

for $0 \leq \gamma \leq 1$. Since $g(0) = 0$, it is sufficient to verify that $g\left(1 - \frac{1}{a}\right) \geq 0$ to complete the proof:

$$\begin{aligned} g\left(1 - \frac{1}{a}\right) &= 1 - \left(1 - \frac{1}{a}\right) - e^{-a\left(1-\frac{1}{a}\right)} \\ &\geq e^{1-a}\left(\frac{e^{a-1}}{a} - 1\right) \\ &\geq e^{1-a}\left(\frac{1+(a-1)}{a} - 1\right) \\ &= 0, \end{aligned}$$

where in the last step we use the inequality $e^x \geq 1 + x$ valid for $x \geq 0$. This completes the proof of Lemma 37. □

Remark 38. In the jigsaw percolation process for $r = 2$ colors (see [6]), the inequality $1 - x \geq e^{-ax}$ is only needed for $0 \leq x \leq 0.9$. In this case it is sufficient to choose the constant $a = 3$. In the general case, if r large (but constant) we have that x is very close to 1 from below, this means that we must take a large.

Using independence and the lower bounds of Lemmas 36 and 37, we can prove the main Lemma of Part I:

Lemma 39. $\mathbf{G}^{(1)}$ *contains a percolating subset of size* $(\log n)^{1+\varepsilon_r}$ *with probability at least* $1 - e^{-\sqrt{n}}$.

Proof. Let $k \leq n/(2(\log n)^{1+\varepsilon_r})$. Applying Lemmas 36 and 37, the probability that in round k we find a percolating subset of size $(\log n)^{1+\varepsilon_r}$ is at least:

$$n^{-O(1)/c_r} \cdot n^{-O(1)/c_r} = n^{-O(1)/c_r}.$$

We conclude that the probability of not finding a percolating subset of size $t_1 = (\log n)^{1+\varepsilon_r}$ in each of the $n/\left(2(\log n)^{1+\varepsilon_r}\right)$ rounds is at most:

$$\left(1 - n^{-O(1)/c_r}\right)^{\frac{n}{2(\log n)^{1+\varepsilon_r}}} \leq \exp\left\{-\frac{n^{1-O(1)/c_r}}{2(\log n)^{1+\varepsilon_r}}\right\} \leq \exp(-\sqrt{n}),$$

where for the first inequality we use the inequality $(1-x)^t \leq e^{-tx}$, valid for $t \geq 0$ and $x \leq 1$. These inequalities hold provided c_r is large enough and n is large enough compared to c_r. This completes the proof of Lemma 39. □

3.2 Part II

In this subsection we aim to prove that conditioned on the existence of a percolating set of size t_1 in $G^{(1)}$, whp there is a percolating set of size at least $\frac{n}{2^{r+2}}$ in $G^{(1)} \cup G^{(2)}$.

We will attempt to construct a percolating set of linear size with the following algorithm:

Algorithm 3 (The doubling algorithm)**.** *This algorithm will proceed in a number of steps. The inputs of the algorithm are the r-fold graph $G^{(2)}$ and a percolating subset X_0 with respect to $G^{(1)}$ of size $(\log n)^{1+\varepsilon_r}$ (see Lemma 39).*

Suppose at the start of the t-step there is a percolating subset X_t with respect to $G^{(1)} \cup G^{(2)}$.

- *Let $A_t := V \setminus X_t$ be the set of active vertices.*

(1) At step $t \geq 0$ we reveal all edges of $G^{(2)}$ between A_t and $X_t \setminus X_{t-1}$, where $X_{-1} := \varnothing$. We define:

- $B_t := \{v \in A_t :\ \forall\ i \in [r]$ *there is a* $v_i \in X_t \setminus X_{t-1}$ *such that* $vv_i \in E_i^{(2)}\}$.

In other words, B_t is the set of active vertices joined to $X_t \setminus X_{t-1}$ by an edge of each colour from the second round of exposure.

(2) If $|B_t| < |X_t|$ we STOP. Otherwise, we set:

- $X_{t+1} := X_t \cup B_t$,
- $A_{t+1} := A_t \setminus B_t$.

If $|X_{t+1}| \geq n/2^{r+2}$ then STOP, otherwise go to (1) for step $t+1$.

Definition 40. *We set $b_t := |B_t|$ and $x_t := |X_t|$ for all t.*

Remark 41. (i) If we reach step $t+1$ in Algorithm 3 then $b_i \geq x_i$ for every $i \in [t]$, therefore:

$$x_i = x_{i-1} + b_{i-1} \geq 2x_{i-1} \text{ for every } i \in [t+1].$$

Thus $b_t \geq x_t \geq 2x_{t-1} \geq 2^2 x_{t-2} \geq ... \geq 2^t x_0 = 2^t t_1$.

(ii) If we reach step $t+1$, then $x_{t+1} = b_t + x_t \leq 2b_t$.

(iii) For the final stopping condition we could have used αn for any $\alpha \in (0, \frac{1}{2})$ instead of $n/2^{r+2}$, but this last expression simplifies the calculations in the subsequent auxiliary lemmas, in particular Lemma 47 Case 1.

(iv) The name of Algorithm 3 comes from [6] where the size of X_t is only doubled in each step. By adding all of B_t rather than some subset of size $|X_0|$ we may more than double the size of X_t in each step, but we consider this a more natural choice for the algorithm. A minor inconvenience is that we no longer know exactly how large X_i is, but a lower bound will be sufficient.

Definition 42. *Let* $t_2 := \max\left\{t \in \mathbb{N} \cup \{0\} : x_t < \frac{n}{2^{r+2}}\right\}$. t_2

Observation 5. *If Algorithm 3 constructs a percolating set X_t of size $\geq \frac{n}{2^{r+2}}$, then it will stop at time $t = t_2 + 1$, otherwise it will stop at time t_2.*

Observation 6. *By Remark 41.(i), we know that $2^{t_2} \leq \frac{b_{t_2}}{t_1}$. It is clear that $b_t \leq n$ for all t, therefore $\log_2 n \geq \log_2\left(\frac{b_{t_2}}{t_1}\right) \geq t_2$ i.e $t_2 = O(logn)$.*

Given an r-fold graph $\boldsymbol{G}$, we denote the event that V contains a percolating subset of size at least m by $\mathcal{P}(\mathbf{G}, m)$. Now we are ready to formally state the main lemma of this subsection: $\mathcal{P}(\mathbf{G}, m)$

Lemma 43. *Conditioned on $G^{(1)}$ containing a percolating subset of size at least $t_1 = (\log n)^{1+\varepsilon_r}$, the probability that $G^{(1)} \cup G^{(2)}$ contains a percolating subset of size at least $n/2^{r+2}$ is at least $1 - \exp\{-t_1/5\}$ for n large enough. Formally written:*

$$\mathbb{P}\left[\mathcal{P}\left(G^{(1)} \cup G^{(2)}, n/2^{r+2}\right) \middle| \mathcal{P}\left(G^{(1)}, (\log n)^{1+\varepsilon_r}\right)\right] \geq 1 - \exp\{-t_1/5\},$$

for n large enough.

The main idea to prove Lemma 43 will be to prove a lower bound on the conditional probability that Algorithm 3 proceeds to step $t+1$ conditioned on it getting to step t. We then multiply these conditional probabilities to obtain a lower bound on the event of finding a percolating subset of size at least $n/2^{r+2}$. We summarize this in the following auxiliary Lemma:

Lemma 44. *Let $t \leq t_2$. Conditioned on the doubling algorithm reaching the t-th step (i.e. X_t is defined and therefore non-empty), the probability that the event $\{b_t \geq x_t\}$ is at least $1 - \exp\{-t_1/4\}$. Formally written:*

$$\mathbb{P}\left[b_t \geq x_t | X_t \neq \varnothing\right] \geq 1 - \exp\{-t_1/4\}.$$

During the proof of Lemmas 43 and 44, we will use some auxiliary claims that will be stated inside the proofs of the lemmas. The proofs of these claims will be left to the end of the section to avoid interrupting the flow of the argument. We will also use the following (one-sided) Chernoff bound (see e.g. [16]):

Lemma 45 (Chernoff Bound). *Let $q \in [0,1]$ and $m \in \mathbb{N}$ and $Z = \sum_{i=1}^{m} Z_i$ where the Z_i's are all independent Bernoulli random variables that take value 1 with probability q and value 0 with probability $1-q$. Let $\mu = \mathbb{E}(Z) = mq$. Then:*

- ***Lower Tail:*** $\mathbb{P}[Z \leq (1-\delta)\mu] \leq \exp\{-\frac{\mu\delta^2}{2}\}$ *for all* $0 < \delta < 1$.

Remark 46. We will not need the corresponding bound on the upper tail probability.

Proof of Lemma 44 . For $t \leq t_2$, the trial set X_t is of size at most $n/2^{r+2}$ (see Definition 42) This means that there are at least $n - n/2^{r+2} \geq n/2$ vertices in the set of active vertices A_t.

We note that the events that individual vertices are in B_t are independent. We define Z_t to be a random variable with binomial distribution $\text{Bi}(|A_t|, q_{t,1}q_{t,2}...q_{t,r})$, where $q_{t,i}$ is the probability that a vertex $v \in A_t$ is joined to $B_{t-1} = X_t \setminus X_{t-1}$ by at least one edge of $G_i^{(2)}$. Then Z_t represents b_t. We state the following auxiliary claims:

Claim 47. *We have:*

$$q_{t,i} \geq \begin{cases} p_i x_t/4 & \text{if } p_i x_t < 2, \\ \\ 1/2 & \text{otherwise.} \end{cases}$$

Claim 48. *Let $t \leq t_2$. Then:*

$$\mathbb{E}[Z_t] \geq 2x_t.$$

From Claim 48 we deduce that:

$$\mathbb{P}\left[b_t \geq x_t | X_t \neq \varnothing\right] \geq \mathbb{P}[Z_t \geq x_t] \geq \mathbb{P}\left[Z_t > \frac{\mathbb{E}[Z_t]}{2}\right].$$

Using Lemma 45 (Chernoff Bound) for $\delta = 1/2$ and Claim 48, we get:

$$\mathbb{P}\left[b_t \geq x_t | X_t \neq \varnothing\right] \geq 1 - \exp\{-\mathbb{E}[Z_t]/8\} \geq 1 - \exp\{-x_t/4\} \geq 1 - \exp\{-x_0/4\}.$$

We recall that $x_0 = t_1$. This completes the proof of Lemma 44. □

We apply Lemma 44 multiple times to obtain Lemma 43.

Proof of Lemma 43. We want to prove that:

$$\mathbb{P}\left[\mathcal{P}\left(G^{(1)} \cup G^{(2)}, n/2^{r+2}\right) \middle| \mathcal{P}\left(G^{(1)}, t_1\right)\right] \geq 1 - \exp\{-t_1/5\}.$$

From Observation 6 we have $t_2 = O(\log n)$. We deduce from Lemma 44 that:

$$\begin{aligned} \mathbb{P}\left[\mathcal{P}\left(G^{(1)} \cup G^{(2)}, n/2^{r+2}\right) \middle| \mathcal{P}\left(G^{(1)}, (\log n)^{1+\varepsilon_r}\right)\right] &\geq \prod_{t=0}^{t_2} \mathbb{P}\left[b_t \geq x_t \middle| X_t \neq \varnothing\right] \\ &\geq \left(1 - \exp\{-t_1/4\}\right)^{O(\log n)}. \end{aligned}$$

Let us note that $(1-x)^t \geq 1 - tx$ for $x \in [0,1]$ and $t \geq 1$. To prove this, we consider the functions $f(x) = (1-x)^t$ and $g(x) = 1 - tx$. We note that $f'(x) = -t(1-x)^{t-1} \geq g'(x) = -t$, for $x \in [0,1]$ and $t \geq 1$. Since $f(0) = g(0)$ and $f'(x) \geq g'(x)$, we conclude that $f(x) \geq g(x)$ in the interval $[0,1]$.

Recalling that $t_1 = (\log n)^{1+\varepsilon_r}$, we can use the previous inequality and the fact that n large enough to conclude that:

$$\begin{aligned} \mathbb{P}\left[\mathcal{P}\left(G^{(1)} \cup G^{(2)}, n/2^{r+2}\right) \middle| \mathcal{P}\left(G^{(1)}, t_1\right)\right] &\geq 1 - O(\log n)\exp\{-t_1/4\} \\ &\geq 1 - \exp\{\log(O(\log n)) - t_1/4\} \\ &\geq 1 - \exp\{-t_1/5\}, \end{aligned}$$

where the last inequality is valid since $t_1 = \Omega(\log n)$. This completes the proof of Lemma 43. □

Thus, we have proved everything we want for Part II except for the Claims 47 and 48.

Proof of Claim 47. Recall that $q_{t,i}$ is the probability that a vertex $v \in A_t$ is joined to $B_{t-1} = X_t \setminus X_{t-1}$ by at least one edge of the second round of exposure $G_i^{(2)}$. From Remark 41.(ii) we know $b_{t-1} \geq x_t/2$ for $0 \leq t \leq t_2$ where $b_{-1} := x_0$, we get:

$$q_{t,i} = 1 - (1-p_i)^{b_{t-1}} \geq 1 - (1-p_i)^{x_t/2} \geq 1 - \exp\{-p_i x_t/2\}. \qquad (1)$$

We now have have two cases:

Case 1: $p_i x_t < 2$. We apply to the right hand side of equation 1 the inequality $1 - e^{-x} \geq x/2$ (valid for $0 \leq x \leq 1$) to get the lower bound $q_{t,i} \geq p_i x_t/4$.

Case 2: $p_i x_t \geq 2$. In this case we just need to use the right hand side of equation 1 and observe that $1 - \exp\{-p_i x_t/2\} \geq 1 - \exp\{-1\} \geq 1/2$ to get the required lower bound $q_{t,i} \geq 1/2$. □

Proof of Claim 48. Let $j_t = \max\{j \in [r] \cup \{0\} : p_j x_t < 2\} \geq 0$, where $p_0 := 0$. Recalling that $A_t \geq n/2$ for $t \leq t_2$, we get:

$$\mathbb{E}[Z] = |A_t| \left(\prod_{j=1}^{r} q_{t,j} \right) \geq \frac{n}{2} \left(\prod_{j=1}^{j_t} q_{t,j} \right) \left(\prod_{j=j_t+1}^{r} q_{t,j} \right).$$

We divide the proof into cases and apply Claim 47:

Case 1: $j_t = 0$. Then by definition of j_t we have $2 \leq p_j x_t$ for all $i \in [j]$. Using Claim 47 we get:

$$\mathbb{E}[Z] \geq \frac{n}{2} \left(\frac{1}{2} \right)^r = 2 \left(\frac{n}{2^{r+2}} \right) \geq 2x_t.$$

Case 2: $j_t \geq 1$. Using Claim 47 and noting that $p_1 x_t \leq p_2 x_t \leq ... \leq p_{j_t} x_t < 2$, we get:

$$\mathbb{E}[Z] \geq \frac{n}{2} \left(\prod_{j=1}^{j_t} \frac{p_j x_t}{4} \right) \left(\frac{1}{2} \right)^{r-j_t} = n x_t^{j_t} P_{j_t} \left(\frac{1}{2} \right)^{r+j_t+1}.$$

Making another case distintion we get:

Case 2.1: $j_t = 1$. We recall that $P_1 = p_1 \geq \frac{c_r^{r-1} \log n}{n}$, thus for n large enough we have:

$$\mathbb{E}[Z] \geq \left(\frac{c_r^{r-1} \log n}{2^{r+2}} \right) x_t \geq 2x_t.$$

Case 2.2: $j_t \geq 2$. We recall that $P_i = p_1 ... p_i \geq c_r^{r-1}/(n \log^{i-1} n)$ for all $2 \leq i \in [r]$ and $x_t \geq 2^t t_1$ for all $0 \leq t \leq t_2$ (see Remark 41 .(i)). Thus:

$$\begin{aligned}\mathbb{E}[Z] \geq \frac{n}{2^{r+j_t+1}} (2^t t_1)^{j_t - 1} x_t \left(\frac{c_{r-1}}{n \log^{j_t - 1} n} \right) &= c_r^{r-1} 2^{t(j_t-1)-r-j_t-1} \left(\frac{t_1}{\log n} \right)^{j_t-1} x_t \\ &\geq c_r^{r-1} 2^{-2r-1} x_t \geq 2x_t,\end{aligned}$$

where the last inequality is valid for c_r large enough. □

3.3 Part III

In this last part we are going to prove that $\boldsymbol{G}^* := G^{(1)} \cup G^{(2)} \cup G^{(3)}$ percolates whp. In more detail:

Lemma 49. *Conditioned on $G^{(1)} \cup G^{(2)}$ containing a percolating subset X of size at least $n/2^{r+2}$, $\mathbf{G}^*$ percolates whp.*

Indeed, we will prove that whp every vertex in $V \setminus X$ is connected to X by edges of every color by using the final round of exposure $G^{(3)}$.

Proof of Lemma 49. We begin by defining K, the event that there is at least one vertex $x \in V \setminus X$ such that at least one color is *not* contained in the set of edges of $G^{(3)}$ between X and x (i.e. $|\{i \in [r] :$ there is $e \in E(G_i^{(3)})$ such that $x \in e\}| < r$). Thus:

$$\mathbb{P}[K] \leq \sum_{i\in[r]} \left(\sum_{v \in V\setminus X} (1-p_i)^{|X|} \right) \leq rn(1-p_1)^{n/2^{r+2}}$$
$$\leq rne^{-p_1 \frac{n}{2^{r+2}}}.$$

We now use the hypothesis that $p_1 \geq (c_r^{r-1} \log n)/n$. We get:

$$\mathbb{P}[K] \leq rne^{-\frac{c_r^{r-1}}{2^{r+2}} \log n} = \frac{rn}{n^{c_r^{r-1}/2^{r+2}}} \leq \frac{r}{n} = o(1),$$

where the last inequality holds provided that $c_r^{r-1} \geq 2^{r+3}$. This completes the proof of Lemma 49. □

4 Concluding remarks.

In this section we present some open problems related to jigsaw percolation:

• Does the jigsaw percolation process on finitely many random graphs have a sharp threshold? We know that every monotone graph property has a sharp threshold in $\mathcal{G}(n,p)$ (see Theorem 2) but we do not know any similar result on r-fold graphs.

In the case of $r = 1$, percolation of the jigsaw process is equivalent to connectivity (as pointed out in Section 1.6). In the case of connectivity, if one edge is added we can only merge at most a pair of components. This is not true for $r \geq 2$ as can be seen in the example below:

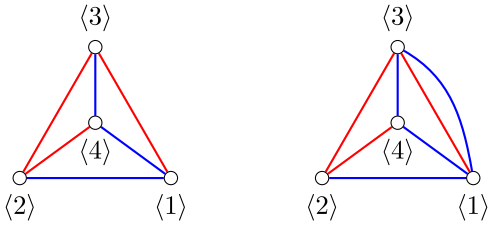


Figure 13: A pair of 2-fold graphs differing in only one edge.

The 2-fold graph on the left hand side does *not* percolate. Even more, its final form after the jigsaw process is the graph itself because no two clusters merge. This is because no two clusters share a blue and red edge. On the other hand, the r-fold graph on the right hand side *does* percolate even though it differs in only one edge from the previous r-fold graph. We note that the clusters $\langle 1\rangle$ and $\langle 3\rangle$ merge to form the cluster $\langle 1,3\rangle$ because they now share a red and blue edge, then in the next step the cluster $\langle 2\rangle$ merges with the cluster $\langle 1,3\rangle$ to form the cluster $\langle 1,2,3\rangle$. In the final step of the jigsaw process, the clusters $\langle 4\rangle$ and $\langle 1,2,3\rangle$ merge to form the cluster $\langle 1,2,3,4\rangle$ concluding the jigsaw process. Therefore, adding one edge can cause the jigsaw process to "snowball" in the sense that merging two clusters (due to the newly added edge) can cause additional clusters to merge in the course of several steps of the jigsaw algorithm. This is not true for connectivity (i.e. the $r = 1$ percolation process).

This suggests that the jigsaw percolation process (for $r \geq 2$) can "explode" even faster than the connectivity process, and since the connectivity process *does* have a sharp threshold, this suggests that the jigsaw percolation process also has a sharp threshold for $r \geq 2$.

• If there is a sharp threshold, what is its exact location? If there is a sharp threshold, we can deduce from Theorem 19 that it should lie between $\frac{T(n)}{C_r}$ and $C_r T(n)$, where $T(n) := \frac{1}{n \log^{r-1} n}$ but we would like to determine the exact constant.

• Let $\boldsymbol{G}$ be an r-fold graph satisfying the conditions of Theorem 19 (ii) (so percolation whp). We always considered r constant but can we let $r(n) \xrightarrow{n\to\infty} \infty$

(i.e the number of colors tending to infinity), while preserving the property $\mathbb{P}[\boldsymbol{G} \text{ percolates}] \xrightarrow{n\to\infty} 1$ and if so, how fast?

If we go through the proof of Theorem 19 and analyse the conditions on c_r and r, we conclude that it will still hold if we let $r \to \infty$ with the following requirements:

$$r = o(\sqrt{\log\log n}),$$
$$C_r \geq 3^{2r^2+r-2} 2^{2r(r-1)}.$$

We would like to remark that there are several requirements that arise during the proof of Theorem 19 but these two requirements are the strongest i.e. they imply all other requirements. In particular, the first requirement is a sufficient condition to guarantee $p_2 t_1 = o(1)$ (See Observation 2), while the second requirement arises from Lemma 39. It is also worth noting that these requirements arise from Theorem 19 (ii) since Theorem 19 (i) would be valid provided that the weaker condition $C_r \geq 2^{r-1}e^3$ holds.

These conditions are rough estimates and we do not know the optimal conditions on r and C_r.

• Another open question is the following: in the case where an r-fold random graph $\boldsymbol{G}(n, p_1, ..., p_r)$ satisfies the conditions of Theorem 19 (ii) with $P_r = \frac{C_r}{n \log^{r-1} n}$, how many steps does it take (asymptotically) for the jigsaw process (see Algorithm 1) to finish? One trivial upper bound is $n-1$. However, analyzing the proof (see below) we can also argue that the number of steps is of order $O((\log n)^{1+\varepsilon_r})$ and by optimizing the proof, we could choose ε_r to be arbitrarily small. However, we do not know if $O(\log n)$ is an upper bound and in the case it is an upper bound we do not know if it is optimal. We now justify the upper bound $O((\log n)^{1+\varepsilon_r})$:

Let $\boldsymbol{G}$ be an r-fold random graph that satisfies conditions of Theorem 19 (ii) with $P_r = \frac{C_r}{n \log^{r-1} n}$. Let $S_{\text{proof}} = t_1 + t_2 + 1$, where $t_1 = (\log n)^{1+\varepsilon_r}$ and $t_2 = O(\log n)$. Let S_{jigsaw} be the number of steps that the jigsaw process run in $\mathbf{G}$ takes to finish. We can prove that $S_{\text{jigsaw}} \leq S_{\text{proof}}$ as follows:

- $*$ From Part I of the proof (see Section 3.1) we know that whp Algorithm 2 constructs a percolating set of size $t_1 = (\log n)^{1+\varepsilon_r}$ by starting with a single well-chosen vertex and adding one vertex at a time. This means that whp the jigsaw process constructs a percolating set of size t_1 in at most t_1 steps, where ε_r is any constant in the interval $\left(0, \frac{r}{r-1}\right)$.
- $*$ From Part II (see Section 3.2) we know that given the percolating set of size t_1 of Part I, then whp Algorithm 3 constructs a percolating set of linear size $\frac{n}{2^{r+2}}$ by doubling the number of vertices that the previous percolating set had. This means that given a percolating set of size t_1 whp the jigsaw process constructs a percolating set of linear size $\frac{n}{2^{r+2}}$ in at most t_2 steps, where $t_2 = O(\log n)$ due to Observation 6.
- $*$ By Part III (see Section 3.3), we know that given the percolating set of size $\Theta(n)$ of Part II, whp the rest of the vertices will share edges of each

color with this percolating set i.e. we can add them all in one step to the percolating set.

It is clear that the sum of the number of steps that each part of proof takes, is an upper bound for the total number of steps of the jigsaw process run in $\mathbf{G}$. Therefore we conclude that:

$$S_{\text{jigsaw}} \leq S_{\text{proof}} \leq (\log n)^{1+\varepsilon_r} + O(\log n) + 1 = O\left((\log n)^{1+\varepsilon_r}\right),$$

i.e. whp $S_{\text{jigsaw}} = O\left((\log n)^{1+\varepsilon_r}\right)$ for every $\varepsilon_r \in \left(0, \frac{r}{r-1}\right)$.

- Let $C_{\text{opt}} = \inf\{C \in \mathbb{R}_+ : \mathbf{G}(n, p_1, p_2, ..., p_r)$ percolates whp, satisfies the conditions of Theorem 19 (ii) with the constant C and $P_r = \frac{C}{n \log^{r-1} n}\}$. We believe that C_{opt} is much smaller than the C_r that we find in the proof, since we did not make any effort to optimize C_r. Let $\varepsilon > 0$ and $\mathbf{G}_1(n, p_1, ..., p_r)$ be an r-fold random graph that satisfies conditions of Theorem 19 (ii) with $P_r = \frac{C_{opt}+\varepsilon}{n \log^{r-1} n}$. It is true even for this P_r that $S_{\text{jigsaw},\mathbf{G}_1} = O(\log n)$ (with the multiplicative constant potentially depending on ε)?

References

[1] N. Alon, J. H. Spencer, *The probabilistic method* (Second Edition), John Wiley & Sons, (2000).

[2] J. A. Baglivo, J. E. Graver, *Incidence and Symmetry in Design and Architecture*, Cambridge University Press, (1983).

[3] N. Biggs, E. Lloyd, R. Wilson, *Graph Theory 1736-1936*, Oxford University Press, (1986).

[4] B. Bollobás, *The evolution of random graphs*, Transactions of the American Mathematical Society, Vol. 286, Num. 1 (1984).

[5] B. Bollobás, O. Cooley, M. Kang and C. Koch, *Jigsaw percolation on random hypergraphs*, arxiv:1603.07883.

[6] B. Bollobás, O. Riordan, E. Slivken, P. Smith, *The threshold for jigsaw percolation on random graphs.*, arxiv:1503.05186v2.

[7] C. D. Brummitt, S. Chatterjee, P. S. Dey, and D.Sivakoff, *Jigsaw percolation: What social networks can collaboratively solve a puzzle?*, The Annals of Applied Probability, Vol. 25, Num. 4 (2015), pp. 2013-2038.

[8] S. Das, *A brief note on estimates of binomial coefficients*, online notes of the Freie Universität Berlin, (2015).

[9] R. Diestel (2000), *Graph Theory*, Springer Verlag New York, (2000).

[10] L. Euler, *Solutio problematis ad geometriam situs pertinentis*, Commentarii academiae scientiarum imperialis Petropolitanae, (1736).

[11] P. Erdős, A. Rényi, *On random graphs*, Publicationes Mathematicae Debrecen, Vol. 6 (1959), pp. 290-297.

[12] P. Erdős, A. Rényi, *On the evolution of random graphs*, Publ. Math. Inst. Hungar. Acad. Sci. 5 (1960), pp. 17-61.

[13] E. Friedgut, G. Kalai, *Every monotone graph property has a sharp threshold*, Proceedings of the American Mathematical Society, Vol. 124, Num. 10 (1996).

[14] A. Frieze, M. Karónski, *Introduction to random graphs*, Cambridge University Press, (2000).

[15] E. N. Gilbert, *Random graphs*, Annals of Mathematical Statistic, Vol. 30, Num. 4 (1959), pp. 1141-1144.

[16] M. Molloy, B. Reed, *Graph Colouring and the Probabilistic Method*, Springer, (2002).

[17] R. Hanneman, M. Riddle, *Introduction to social networks methods*, free introductory text book on social network analysis, (2005).

[18] M. Harré, *Social Network Size Linked to Brain Size*, https://www.scientificamerican.com/article/social-network-size-linked-brain-size/ , (2012).

[19] J. L. Powell, P. A. Lewis, R. I. M. Dunbar, M. García-Fiñana, N. Roberts, *Orbital prefrontal cortex volume correlates with social cognitive competence*, Procedings of the Royal Society B, Vol. 48, Issue 12 (2010), pp. 3554–3562.

[20] D. Smith, *Proof! Just six degrees of separation between us*, The Guardian, Sunday 7 August (2008).