The threshold for jigsaw percolation on random graphs

Béla Bollobás

Department of Pure Mathematics and Mathematical Statistics
University of Cambridge, U.K.

Department of Mathematical Sciences
University of Memphis, U.S.A.

London Institute for Mathematical Sciences
35a South Street, London W1K 2XF, U.K.

`b.bollobas@dpmms.cam.ac.uk`

Oliver Riordan

Mathematical Institute
University of Oxford, U.K.

`riordan@maths.ox.ac.uk`

Erik Slivken

Department of Mathematics
UC Davis, U.S.A.

`erikslivken@math.ucdavis.edu`

Paul Smith

Department of Pure Mathematics and Mathematical Statistics
University of Cambridge, U.K.

`p.j.smith@dpmms.cam.ac.uk`

Abstract

Jigsaw percolation is a model for the process of solving puzzles within a social network, which was recently proposed by Brummitt, Chatterjee, Dey and Sivakoff. In the model there are two graphs on a single vertex set (the 'people' graph and the 'puzzle' graph), and vertices merge to form components if they are joined by an edge of each graph. These components then merge to form larger components if again there is an edge of each graph joining them, and so on. Percolation is said to occur if the process terminates with a single component containing every vertex. In this note we determine the threshold for percolation up to a constant factor, in the case where both graphs are Erdős–Rényi random graphs.

1   Introduction

Jigsaw percolation is a dynamical percolation model on finite graphs, which was proposed by Brummitt, Chatterjee, Dey and Sivakoff [2] as a tool for the study of sequences of interactions within a social network that enable a group of individuals to collectively solve a problem. In the model there are two edge sets defined on a common set of vertices, and, at discrete times, clusters of vertices merge to form larger clusters if they are joined by at least one edge of each type. Before expanding on the motivation for the model, let us give the formal definition. We write $[n]$ for $\{1, 2, \ldots, n\}$.

Definition 1. For $i = 1, 2$, let $E_i \subset [n]^{(2)}$ be a set of pairs of elements of $V := [n]$. Let G be the ordered triple $G := (V, E_1, E_2)$; we call this object a *double graph*. *Jigsaw percolation* with input G evolves at discrete times $t = 0, 1, \ldots$ according to the following algorithm. At time t there is a partition $\mathcal{C}_t = \{C_t^1, \ldots, C_t^{k_t}\}$ of the vertex set $[n]$, which is constructed inductively as follows:

(1) We take $k_0 = n$ and $C_0^i = \{i\}$ for all $1 \leqslant i \leqslant n$. That is, at time 0 we begin with every vertex in a separate set of the partition.

(2) At time $t \geqslant 0$, construct a graph $\mathcal{G}_t$ on vertex set $\mathcal{C}_t$ by joining C_t^i to C_t^j if there exist edges $e_1 \in E_1$ and $e_2 \in E_2$ such that

$$e_\ell \cap C_t^k \neq \emptyset$$

for each of the four choices of $\ell \in \{1, 2\}$ and $k \in \{i, j\}$.

(3) If $E(\mathcal{G}_t) = \emptyset$, then STOP. Otherwise, construct the partition

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

corresponding to the connected components of $\mathcal{G}_t$, so each part C_{t+1}^i is a union of those parts of $\mathcal{C}_t$ corresponding to a component of $\mathcal{G}_t$.

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

Since $|\mathcal{C}_t|$ is strictly decreasing, the algorithm terminates in time at most $\binom{n}{2}$. We denote the final partition by $\mathcal{C}_\infty = (C_\infty^1, \ldots, C_\infty^{k_\infty})$. We say that there is *percolation*, or that the double graph is *solved*, if $\mathcal{C}_\infty = \{V\}$, i.e., if we stop in step (4).

Less formally, the jigsaw percolation algorithm begins with each vertex considered a separate cluster, and proceeds by merging, at each step, clusters of vertices joined by at least one edge from E_1 and at least one edge from E_2.

Let us mention in passing a superficially similar, but very different, percolation model in double graphs introduced by Buldyrev, Parshani, Paul, Stanley and Havlin [3] in 2010. The set-up is the same, but one defines the partition in a top-down way, finding the maximal sets of vertices connected in both graphs (V, E_1) and (V, E_2) ('mutually connected clusters'). To see the difference note that if these graphs are edge-disjoint and

connected, then in this model V forms a single 'mutually connected cluster', whereas in jigsaw percolation the algorithm stops where it starts, with a partition into singletons.

Returning to the jigsaw model, which we consider throughout this paper, Brummitt, Chatterjee, Dey and Sivakoff [2] suggest that jigsaw percolation may be a suitable model for analysing how a puzzle may be solved by collaboration between individuals in a social network. The premise is that each individual has a 'piece' of the puzzle, and that these 'pieces' must be combined in a certain way in order to solve the puzzle. The process of solving the puzzle is constrained by the social network of the individuals concerned. To model this, the authors of [2] suggest that one of the graphs, $G_1 := (V, E_1)$ say, (which they call the *people graph*, and which we call the *red graph*), could represent the graph of acquaintances, and that the other graph, $G_2 := (V, E_2)$, (which they call the *puzzle graph*, and which we call the *blue graph*), could represent the 'compatibility' between pairs of 'pieces' of the puzzle. The jigsaw percolation algorithm thus represents the merging of 'compatible puzzle pieces' by groups of connected individuals. For an in-depth account of the applications of the model to social networks, we refer the reader to the original article [2].

In [2], the authors prove a number of necessary and sufficient conditions for percolation when the red graph G_1 is the Erdős–Rényi random graph $\mathcal{G}(n, p)$ with n vertices and edge probability p and the blue graph G_2 is a deterministic graph, such as the n-cycle or another graph of bounded maximum degree. For example, they show that there is an absolute constant $c > 0$ such that if, for each $n \in \mathbb{N}$, $G_2^n = ([n], E_2^n)$ is an arbitrary connected graph on vertex set $[n]$, and G_1^n is an Erdős–Rényi graph with edge probability $p \geqslant c / \log n$, then the double graph $G := ([n], E_1^n, E_2^n)$ percolates with high probability as $n \to \infty$. On the other hand, if the graphs G_2^n have bounded maximum degree and instead $p < n^{-\varepsilon}$ for some $\varepsilon > 0$, then with high probability the double graph does not percolate.

Gravner and Sivakoff [5] observe that for certain deterministic graphs G_2, the jigsaw percolation model behaves similarly to bootstrap percolation on the grid $[n]^2$, and they use techniques from bootstrap percolation to prove tight bounds in certain cases. For example, if $G_1 = \mathcal{G}(n, p)$ and $G_2 = C_n$ is the n-cycle, they show that the *critical probability* $p_c^{C_n}(n)$ for the corresponding double graph G, defined by

$$p_c^{C_n}(n) := \inf \big\{ p \,:\, \mathbb{P}(G \text{ percolates}) \geqslant 1/2 \big\},$$

satisfies

$$p_c^{C_n} = \frac{(1 + o(1))\pi^2/6}{\log n}.$$

(This critical probability, and specifically the constant $\pi^2/6$, will be known to readers who are familiar with bootstrap percolation: it is also (see [6]) the critical probability for the so-called 'modified' bootstrap percolation model on $[n]^2$ – this is, of course, not a coincidence (see [5] for the details).)

In this note we study the case where *both* underlying graphs are Erdős–Rényi random graphs. In order to state our result, we need a little more notation. For the rest of the paper, we shall take $G_1 = (V, E_1)$ and $G_2 = (V, E_2)$ to be independent Erdős–Rényi random graphs with the same vertex set $V = [n]$, with edge probabilities p_1 and

p_2 respectively, and we take $G = ([n], E_1, E_2)$. A first trivial observation is that if the double graph G is to percolate then both G_1 and G_2 must be connected. We shall ensure this by assuming

$$\min\{p_1, p_2\} \geqslant \frac{c \log n}{n}, \tag{1}$$

for a sufficiently large absolute constant c.[1] Under this condition, in this note we determine the critical value $p_c(n)$ of the product $p_1 p_2$ up to a constant factor. More precisely, we show that under the assumption (1), if $p_1 p_2 \leqslant (1/c) p_c(n)$ then percolation is very unlikely, and if $p_1 p_2 \geqslant c p_c(n)$ then percolation is very likely.

Theorem 2. *There is an absolute constant $c > 0$ such that the following holds, with $G = (V, E_1, E_2)$, where $(V, E_1) = \mathcal{G}(n, p_1)$ and $(V, E_2) = \mathcal{G}(n, p_2)$ are independent Erdős–Rényi random graphs on the same set of n vertices, and p_1 and p_2 are functions of n.*

(i) *If $p_1 p_2 \leqslant 1/(cn \log n)$, then $\mathbb{P}(G \text{ percolates}) \to 0$.*

(ii) *If $p_1 p_2 \geqslant c/(n \log n)$ and (1) holds, then $\mathbb{P}(G \text{ percolates}) \to 1$.*

Informally, this result says that

$$p_c(n) = \Theta\left(\frac{1}{n \log n}\right).$$

If p_2 satisfying (1) is given, then another way of thinking about Theorem 2 is that the critical value of p_1 is $\Theta(1/p_2 n \log n) = \Theta(1/d \log n)$ (provided this also satisfies (1)), where $d = (n-1)p_2$ is the expected degree of vertices in G_2. Theorems 1 and 2 of [5] also show that, for certain families of (deterministic) d-regular graphs G_2 (with d possibly being a function of n), the critical value of p_1 is $\Theta(1/d \log n)$, although the authors of [5] also show that this does not always hold.

The proof of lower bound in Theorem 2 is straightforward and follows from standard methods; the real content of this paper is the proof of the upper bound.

2 Proof of part (i) of Theorem 2

Here we present the very brief proof of part (i) of Theorem 2, although really it is no more than the argument used by Aizenman and Lebowitz [1] to derive the lower bound up to a constant factor for the critical probability for 2-neighbour bootstrap percolation on $[n]^2$.

The percolation process defined above can be broken down into smaller steps, in each of which two clusters (parts of the current partition) merge – specifically, one can modify step (3) of Definition 1 to merge an arbitrary pair of sets C_t^i joined in the graph $\mathcal{G}_t$, rather than entire connected components. Since we start with a partition into singletons, considering the first step at which a cluster of size at least $\log n$ appears, it follows that if

---

[1]With some tightening, our arguments could be made to work under the (optimal) assumption that $\min\{p_1, p_2\} \geqslant (1 + \varepsilon) \log n/n$, where $\epsilon > 0$ is fixed but arbitrary. We choose to make the stronger assumption in (1) for clarity of the exposition.

(G, E_1, E_2) percolates, then there is some set A of at least $\log n$ but at most $2\log n$ vertices such that the red and blue graphs restricted to A are both connected. Using independence of the red and blue graphs, the facts that a connected graph must contain a spanning tree and that there are k^{k-2} labelled trees on k vertices, and the bound $\binom{n}{k} \leqslant (en/k)^k$, we see that

$$
\begin{aligned}
\mathbb{P}(G \text{ percolates}) \ &\leqslant\ \sum_{k=\log n}^{2\log n} \binom{n}{k} k^{k-2} p_1^{k-1} k^{k-2} p_2^{k-1} \\
&\leqslant\ \frac{1}{p_1 p_2} \sum_{k=\log n}^{2\log n} (enk p_1 p_2)^k \\
&\leqslant\ \frac{1}{p_1 p_2} \sum_{k=\log n}^{2\log n} (2enp_1 p_2 \log n)^k \\
&\leqslant\ 2en\log n \sum_{k=\log n}^{\infty} (2enp_1 p_2 \log n)^{k-1}.
\end{aligned}
$$

For $p_1 p_2 \leqslant 1/(e^4 n\log n)$, say, the quantity in brackets is at most $1/e^2$ and it follows that the final bound is $o(1)$, proving (i).

We can now move on to the main part of this paper: the second part of Theorem 2.

3 Proof of part (ii) of Theorem 2

Let us begin with a small number of conventions. As already mentioned, G_1 and G_2 will always be independent Erdős–Rényi random graphs on vertex set $V := [n]$, with densities p_1 and p_2 respectively. Throughout, we assume that c is a sufficiently large absolute constant, that the number of vertices n is sufficiently large, and that p_1 and p_2 satisfy

$$
p_1 p_2 = \frac{c}{n\log n} \qquad \text{and} \qquad \frac{c\log n}{n} \leqslant p_1 \leqslant p_2. \tag{2}
$$

The assumptions of Theorem 2 (ii) require $p_1 p_2 \geqslant c/(n\log n)$ rather than the equality in (2), but we may couple with smaller p_1 and p_2 if necessary so that (2) holds. Constants implicit in $O(\cdot)$ notation (and its variants) are independent of c (and of n). For later use, let us note some immediate consequences of (2); these follow since $p_1 \leqslant \sqrt{p_1 p_2}$ and $p_2 = (p_1 p_2)/p_1$:

$$
p_1 \leqslant \left(\frac{c}{n\log n}\right)^{1/2} \leqslant n^{-1/2} \qquad \text{and} \qquad p_2 \leqslant \frac{1}{(\log n)^2}. \tag{3}
$$

We need a key definition: that of an 'internally spanned' set of vertices. The definition enables one to say which sets of sites (internally) percolate, without any help from other vertices, and is motivated by several similar notions in the bootstrap percolation literature

(see, for example, [1, 4]). Our method for showing that G percolates will broadly take the form 'there exists a nested sequence of internally spanned sets $U_1 \subset \cdots \subset U_m$, with $U_m = [n]$'; the crux will be finding such a sequence.

Definition 3. A set $U \subset [n]$ is *internally spanned* by G if the double graph $G^U := (U, E_1^U, E_2^U)$ percolates, where E_i^U is the edge set of the induced subgraph of G_i on vertex set U, for $i = 1, 2$ (that is, $E_i^U := E\big(G_i[U]\big)$). We write $I(G, m)$ for the event that $V(G)$ contains an internally spanned set of size at least m.

The proof of the lower bound of Theorem 2 could now be rephrased as follows. First, observe that if G percolates, then, by merging components two at a time, we have that $V(G)$ must contain an internally spanned set of size roughly $\log n$. Second, using well-known properties of trees, one can show that if p is small then this event is unlikely to occur.

The proof of the upper bound is divided into three parts, with a corresponding division of both the red and blue edges into three subsets. In the first part of the proof we show that with high probability there is a set A of at least $(\log n)^{3/2}$ vertices which is internally spanned by the first set of (red and blue) edges. This is the core of the proof: the 'bottleneck' event to percolation (in a certain sense) is the existence of an internally spanned set of size about $\log n$.[2] Then we show, using the second set of edges, that with high probability the set A is contained in an internally spanned (with respect to the edges reveal so far) set B of size $n/16$. Finally, using the condition (1), we show using the third set of edges and the set B that with high probability the whole vertex set is internally spanned.

In each of the first two parts we specify an 'exploration algorithm', in which the edges of each of the underlying graphs are revealed in an order that depends on what has been observed so far. The purpose is to reveal as few edges as possible (in order that we may reveal them later if necessary) in the search for a nested sequence of internally spanned sets. The algorithms are set out explicitly in Definitions 5 and 10.

Between the three parts of the proof, independence is maintained by *sprinkling*: for each $i = 1, 2$ and $j = 1, 2, 3$, we take $G_i^{(j)}$ to be an independent copy of $\mathcal{G}(n, p_i)$, where p_1 and p_2 satisfy the conditions (2) as before; we then set $E_i^{(j)} := E(G_i^{(j)})$,

$$G_i := G_i^{(1)} \cup G_i^{(2)} \cup G_i^{(3)}, \qquad \text{and} \qquad G^{(j)} := \big([n], E_1^{(j)}, E_2^{(j)}\big).$$

Constructing the G_i in this way maintains both conditions in (2), with a different value of c. More precisely, the edge probability p_i' of G_i satsfies $1 - p_i' = (1 - p_i)^3$, so p_i' is a little less than $3p_i$. One could therefore replace c by $9c$ in (2), and any double graph satisfying the new conditions could be coupled with our double graph.

---

[2]This observation was also exploited in the proof of the lower bound, although it is nothing new: a similar idea was used in [2, 5] on jigsaw percolation, and previously in [1, 6] (among many other papers) on bootstrap percolation.

3.1 Part I

In this first part of the proof we prove the following lemma.

Lemma 4. *The probability that $G^{(1)}$ contains an internally spanned set of size at least $(\log n)^{3/2}$ is at least $1 - e^{-\sqrt{n}}$.*

We prove the lemma by repeatedly attempting to build an internally spanned set of size at least $(\log n)^{3/2}$ by adding one vertex at a time to a so-called 'trial set' (we call this the *1-by-1 algorithm*). If we find a suitable vertex to add to the trial set, then we continue. If not, then we discard the vertices from the trial set, and start again; we call this starting a new *round*. Discarding the trial set will ensure independence between rounds (see below). More precisely, the algorithm performs a sequence of 'tests', asking whether certain potential red edges or potential blue edges are present. (More precisely still, the 'test' corresponding to a pair $\{x, y\}$ of vertices and $i \in \{1, 2\}$ asks whether $xy \in E_i^{(1)}$.) We shall make sure that no test is performed twice.

The subtlety is in the order in which we reveal the edges: the aim is to reveal as few edges as possible in the search for each new vertex. Given an internally spanned trial set X, we first reveal all red edges from (not-yet-discarded) vertices outside X to the *most recently added* vertex v in X. Since a potential red edge is tested immediately after the first time one of its ends is added to X, it cannot be tested twice within a round.[3] Let R be the set of vertices outside X incident with such red edges. We test for blue edges, to the whole of the trial set, *only* from vertices in R. If there is a vertex in R with a blue edge to any vertex in the trial set, then we add one such vertex to the trial set. We discard all other vertices in R until the end of this round; this ensures that no potential blue edge is tested twice within a round. At the end of a round we permanently discard all vertices in the trial set. Since a tested edge (red or blue) always has at least one end in the trial set, this ensures that no edge is tested in two different rounds, giving us the independence we need.

Here is a formal description of the algorithm.

Definition 5. (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 kth round there is a set $A_k \subset [n]$ of active vertices and a set $D_k \subset [n]$ of discarded vertices. We begin with $A_1 = [n]$ and $D_1 = \emptyset$. The procedure for the kth round is as follows:

(1) At the start of the tth step of the kth round there is a set $X_k^t = \{x_k^1, \ldots, x_k^t\} \subset A_k$ of trial vertices, a set $A_k^t \subset A_k$ of active vertices, and a set $D_k^t \subset A_k$ of discarded vertices. These sets partition A_k, so for all t, A_k is the disjoint union of X_k^t, A_k^t and D_k^t. To begin, we have $X_k^0 = D_k^0 = \emptyset$ and $A_k^0 = A_k$.

---

[3] Looking at it from the point of view of vertices, rather than edges, the fact that a potential new vertex has been considered at the tth step, and found not to have a red edge to the most recently added vertex, does not stop us from testing the same vertex again at later steps, since in those steps we will be testing for different red edges.

(2) For $t = 0$, move an arbitrary active vertex to the trial set. That is, set $X_k^1 = \{x_k^1\}$, $D_k^1 = \emptyset$ and $A_k^1 = A_k^0 \setminus \{x_k^1\}$, where $x_k^1 \in A_k^0$ is arbitrary.

(3) For $t \geqslant 1$, reveal all edges of $G_1^{(1)}$ (that is, all red edges from the first sprinkling) between A_k^t and $\{x_k^t\}$, and let

$$R_k^t := \{x \in A_k^t \ : \ xx_k^t \in E_1^{(1)}\}.$$

Then, reveal all edges of $G_2^{(1)}$ (that is, all blue edges from the first sprinkling) between R_k^t and X_k^t, and let

$$B_k^t := \{x \in R_k^t \ : \ xx_k^s \in E_2^{(1)} \text{ for some } 1 \leqslant s \leqslant t\}.$$

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

$$X_k^{t+1} := X_k^t \cup \{x_k^{t+1}\}, \quad A_k^{t+1} := A_k^t \setminus R_k^t, \quad \text{and} \quad D_k^{t+1} := D_k^t \cup R_k^t \setminus \{x_k^{t+1}\}.$$

If $t \geqslant (\log n)^{3/2}$ then STOP, otherwise set $t := t + 1$ and go to step (3).

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

$$A_{k+1} := A_k \setminus X_k \quad \text{and} \quad D_{k+1} := D_k \cup X_k.$$

If

$$k \geqslant \frac{n}{2(\log n)^{3/2}}$$

then STOP, otherwise set $k := k + 1$ and $t := 0$, and go to step (1).

Before starting the analysis, let us note that since we consider at most $n/(2(\log n)^{3/2})$ rounds, and stop each with a trial set of size at most $(\log n)^{3/2}$, we start each round with

$$|A_k^0| = |A_k| \geqslant n/2. \tag{4}$$

Let $\mathcal{B}_k^t$ be the event that X_k^{t+1} is (defined and) has size $t + 1$ (if $t \geqslant 1$ then this is equivalent to the event that B_k^t is (defined and) non-empty). We shall show that $\mathcal{B}_k^t$ is not too unlikely. For technical reasons, we also need to consider the event

$$\mathcal{S}_k^t = \{|R_k^s| \leqslant n^{3/4} \text{ for } s = 1, 2, \ldots, t\}$$

that within round k, we have not 'used up' too many vertices by step t. (Here and in what follows we ignore rounding to integers in expressions such as $n^{3/4}$. This makes essentially no difference.) For $k \leqslant n/(2(\log n)^{3/2})$ and $t \geqslant 1$, let

$$r_k^t := \mathbb{P}\big(\mathcal{B}_k^t \cap \mathcal{S}_k^t \ \big| \ \mathcal{B}_k^{t-1} \cap \mathcal{S}_k^{t-1}\big),$$

noting that $\mathcal{S}_k^0$ is the trivial event that always holds

We shall need two different estimates on r_k^t, according to whether t is larger or smaller than $(\log n)/c$.

Lemma 6. *Suppose that $k \leqslant n/\big(2(\log n)^{3/2}\big)$ and $1 \leqslant t \leqslant t_1 = (\log n)^{3/2}$. Then*

$$
r_k^t \geqslant
\begin{cases}
1 - \exp\big(-(n/5)p_1 p_2 t(1 - p_2 t)\big) & unconditionally, \\
(1/10)n p_1 p_2 t(1 - p_2 t) & if\ n p_1 p_2 t \leqslant 1.
\end{cases}
$$

Proof. We condition on the outcome of the exploration so far, up to the start of step t of round k. Note that this information determines whether $\mathcal{B}_k^{t-1} \cap \mathcal{S}_k^{t-1}$ holds; we may assume that it does.

Given the information revealed so far, the conditional distribution of $|R_k^t|$ is binomial $\mathrm{Bin}(|A_k^t|, p_1)$. Since $|A_k^t| \leqslant n$, we have

$$
\mathbb{P}\big((\mathcal{S}_k^t)^{\mathrm{c}} \mid \mathcal{B}_k^{t-1} \cap \mathcal{S}_k^{t-1}\big) = \mathbb{P}\big(|R_k^t| > n^{3/4} \mid \mathcal{B}_k^{t-1} \cap \mathcal{S}_k^{t-1}\big)
$$
$$
\leqslant \binom{n}{n^{3/4}} p_1^{n^{3/4}} \leqslant \big(en^{1/4}p_1\big)^{n^{3/4}} \leqslant e^{-\sqrt{n}},
$$

say, where for the first inequality we have taken a union bound and for the final inequality we have used the bound $p_1 \leqslant 1/\sqrt{n}$ (from (3)).

Now, conditional on the exploration so far, for each $x \in A_k^t$ we have

$$
\mathbb{P}\big(x \in B_k^t\big) = p_1\big(1 - (1 - p_2)^t\big) \geqslant p_1\big(1 - e^{-p_2 t}\big) \geqslant p_1 p_2 t(1 - p_2 t),
$$

since $1 - e^{-x} \geqslant x(1 - x)$ for all $x \geqslant 0$. On the event $\mathcal{B}_k^{t-1} \cap \mathcal{S}_k^{t-1}$ we have

$$
|A_k^t| \geqslant |A_k^0| - t n^{3/4} \geqslant n/4,
$$

using (4). Since $\mathcal{B}_k^t$ holds if and only if $B_k^t \neq \emptyset$, we thus have

$$
\mathbb{P}\big((\mathcal{B}_k^t)^{\mathrm{c}} \mid \mathcal{B}_k^{t-1} \cap \mathcal{S}_k^{t-1}\big) \leqslant \big(1 - p_1 p_2 t(1 - p_2 t)\big)^{n/4}
$$
$$
\leqslant \exp\big(-(n/4)p_1 p_2 t(1 - p_2 t)\big),
$$

so

$$
r_k^t \geqslant 1 - \exp\big(-(n/4)p_1 p_2 t(1 - p_2 t)\big) - e^{-\sqrt{n}}.
$$

The first case of the lemma now follows from the bounds

$$
n p_1 p_2 t = \frac{ct}{\log n} = o\big(\sqrt{n}\big) \qquad \text{and} \qquad p_2 t \leqslant \frac{(\log n)^{3/2}}{(\log n)^2} = o(1).
$$

The second case follows from the first and the inequality $1 - e^{-x} \geqslant x/2$, valid for $x \leqslant 1$. $\square$

In the next two lemmas, we break down the 1-by-1 algorithm into two stages: first, in Lemma 7, we show that the probability the algorithm reaches step

$$
t_0 := (\log n)/c
$$

in a given round is at least $n^{-O(1)/c}$. Then, in Lemma 8, we show that the probability it reaches step

$$
t_1 := (\log n)^{3/2},
$$

given that it has reached step t_0, is also at least $n^{-O(1)/c}$.

Lemma 7. *Suppose that $k \leqslant n/\big(2(\log n)^{3/2}\big)$. Then*

$$\mathbb{P}\big(\mathcal{B}_k^{t_0} \cap \mathcal{S}_k^{t_0} \mid \mathcal{B}_k^0\big) \geqslant n^{-4/c}.$$

Proof. The definition of t_0 combined with the expression for $p_1 p_2$ from (2) implies that $n p_1 p_2 t_0 \leqslant 1$, and hence the second case of Lemma 6 applies for the whole range. Thus,

$$\mathbb{P}\big(\mathcal{B}_k^{t_0} \cap \mathcal{S}_k^{t_0} \mid \mathcal{B}_k^0\big) \geqslant \prod_{t=1}^{t_0} \frac{1}{10} n p_1 p_2 t (1 - p_2 t) \geqslant \left(\frac{c}{10 \log n}\right)^{t_0} t_0! (1 - p_2 t_0)^{t_0}.$$

Recall that $p_2 \leqslant (\log n)^{-2}$, so $p_2 t_0 \leqslant (\log n)^{-1} = o(1)$. Noting that $1 - x \geqslant e^{-2x}$ if $x \leqslant 1/2$ and that $t! \geqslant (t/e)^t$ for all $t \in \mathbb{N}$, it follows that

$$\mathbb{P}\big(\mathcal{B}_k^{t_0} \cap \mathcal{S}_k^{t_0} \mid \mathcal{B}_k^0\big) \geqslant \left(\frac{c t_0}{10 e \log n}\right)^{t_0} \exp\big(-2 p_2 t_0^2\big) = \exp\big(-t_0 \log(10e) - 2 p_2 t_0^2\big).$$

Since $p_2 t_0 \leqslant 1/\log n$, we obtain

$$\mathbb{P}\big(\mathcal{B}_k^{t_0} \cap \mathcal{S}_k^{t_0} \mid \mathcal{B}_k^0\big) \geqslant \exp(-4 t_0) = n^{-4/c},$$

as required. $\qquad\square$

Lemma 8. *Suppose that $k \leqslant n/\big(2(\log n)^{3/2}\big)$. Then*

$$\mathbb{P}\big(\mathcal{B}_k^{t_1} \mid \mathcal{B}_k^{t_0} \cap \mathcal{S}_k^{t_0}\big) \geqslant n^{-O(1)/c}.$$

Proof. For $t_0 < t \leqslant t_1$, we use the first of the two estimates in Lemma 6, the validity of which does not depend on t. Using this, and recalling that $p_2 \leqslant 1/(\log n)^2$ (from (3)) and $t \leqslant (\log n)^{3/2}$, we have

$$
\begin{aligned}
\mathbb{P}\big(\mathcal{B}_k^{t_1} \mid \mathcal{B}_k^{t_0} \cap \mathcal{S}_k^{t_0}\big) &\geqslant \prod_{t=t_0+1}^{t_1} \Big(1 - \exp\big(-(n/5) p_1 p_2 t(1 - p_2 t)\big)\Big) \\
&\geqslant \prod_{t=t_0}^{t_1} \Big(1 - \exp\big(-(n/6) p_1 p_2 t\big)\Big) \\
&\geqslant \prod_{t=t_0}^{t_1} \Big(1 - \exp\big(-ct/(6 \log n)\big)\Big) \\
&\geqslant \exp\Big(-3 \sum_{t=t_0}^{t_1} \exp\big(-ct/(6 \log n)\big)\Big),
\end{aligned}
$$

where for the final step we used the inequality $1 - x \geqslant \exp(-3x)$, valid (by convexity) for $0 \leqslant x \leqslant 0.9$, say.[4] Thus,

$$\mathbb{P}\big(\mathcal{B}_k^{t_1} \mid \mathcal{B}_k^{t_0} \cap \mathcal{S}_k^{t_0}\big) \geqslant \exp\left(-\frac{3 e^{-1/6}}{1 - e^{-c/(6 \log n)}}\right) \geqslant n^{-O(1)/c},$$

where the implied constant does not depend on c. $\qquad\square$

---

[4]What we really need in this argument is that $p_2 \ll 1/\log n$, or equivalently $p_1 \gg 1/n$, which already implies the existence of a giant component in each colour. We only need the connectivity condition once we have obtained an internally spanned set of linear size.

We now put the previous few lemmas together.

Proof of Lemma 4. Let $k \leqslant n/\big(2(\log n)^{3/2}\big)$. Then in the kth round, the probability of finding an internally spanned set of size $(\log n)^{3/2}$ is at least $n^{-O(1)/c}$, by applying Lemmas 7 and 8 in turn. Moreover, this bound holds conditional on the result of all previous bounds, since in proving Lemma 7 and 8 we conditioned on these previous rounds. Hence the probability that all $n/\big(2(\log n)^{3/2}\big)$ rounds terminate 'early' (without finding an internally spanned set of size $(\log n)^{3/2}$) is at most

$$\left(1 - n^{-O(1)/c}\right)^{n/(2(\log n)^{3/2})} \leqslant \exp\left(-\Omega\left(\frac{n^{1-O(1)/c}}{(\log n)^{3/2}}\right)\right) \leqslant \exp\left(-\sqrt{n}\right),$$

if c is sufficiently large. □

3.2 Part II

In this part of the proof we prove the following lemma.

Lemma 9. *Given that $G^{(1)}$ contains an internally spanned set of size at least $(\log n)^{3/2}$, the conditional probability that $G^{(1)} \cup G^{(2)}$ contains an internally spanned set of size at least $n/16$ is at least $1 - n^{-100}$. That is,*

$$\mathbb{P}\Big(I\big(G^{(1)} \cup G^{(2)}, n/16\big) \ \Big| \ I\big(G^{(1)}, (\log n)^{3/2}\big)\Big) \geqslant 1 - n^{-100}.$$

In this range we use a different vertex exploration algorithm in order to find successively larger internally spanned sets. Rather than adding vertices 1-by-1, as in Part I, we attempt to double the size of the trial set at each step. We start with a set X_0 of size $t_1 = (\log n)^{3/2}$ internally spanned by $G^{(1)}$ (found in Part I), and we continue so that at step t we have a set X_t of size

$$x_t := 2^t t_1$$

internally spanned by $G^{(1)} \cup G^{(2)}$. In order to maintain independence between steps, we only add a new vertex v to the trial set if there is at least one edge of each colour from $G^{(2)}$ joining v to the subset of the trial vertices that was added at the previous step.

Definition 10. (The doubling algorithm.) At the start of the tth step there is a set X_t of vertices internally spanned by $G^{(1)} \cup G^{(2)}$, where $|X_t| = x_t$. The set X_t is the *trial set*. The set $A_t := V(G) \setminus X_t$ of remaining vertices in the graph is the *active set*. The algorithm takes as its inputs the double graphs $G^{(1)}$ and $G^{(2)}$, and a set X_0 of size $(\log n)^{3/2}$, internally spanned by $G^{(1)}$.

(1) At step $t \geqslant 0$, reveal all edges of $G^{(2)}$ between A_t and $X_t \setminus X_{t-1}$, where we set $X_{-1} = \emptyset$. Let

$$B_t := \bigcup_{v', v'' \in X_t \setminus X_{t-1}} \Big\{v \in A_t \ : \ vv' \in E_1^{(2)} \text{ and } vv'' \in E_2^{(2)}\Big\}.$$

Thus, 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 sprinkling.

(2) If $|B_t| \leqslant x_t$ then STOP. Otherwise, let $C_t \subset B_t$ be an arbitrary set of exactly x_t vertices of B_t, and set

$$X_{t+1} := X_t \cup C_t, \qquad \text{and} \qquad A_{t+1} := A_t \setminus C_t.$$

If $|X_{t+1}| \geqslant n/16$ then STOP, otherwise go to step (1).

First we need a lower bound on the probability that the size of the trial set doubles at step t.

Lemma 11. *The probability that $|B_t| \geqslant x_t$, conditional on the doubling algorithm having reached the tth step (that is, $|X_t| = x_t$), is at least*

$$1 - \exp\Big(-\Omega\big(c(\log n)^2\big)\Big).$$

Proof. Let s_t be the probability in question, so

$$s_t = \mathbb{P}\big(|B_t| \geqslant x_t \mid |X_t| = x_t\big).$$

Let $q_{t,i}$ be the probability that a vertex $v \in A_t$ is joined to $X_t \setminus X_{t-1} = C_{t-1}$ by at least one edge of $G_i^{(2)}$. Since $|C_{t-1}| = x_t/2$ for $t \geqslant 1$ while $|C_{-1}| = x_0$, we have

$$q_{t,i} \geqslant 1 - (1 - p_i)^{x_t/2} \geqslant 1 - e^{-p_i x_t/2} \geqslant \begin{cases} p_i x_t/4 & \text{if } p_i x_t < 2, \\ 1/2 & \text{otherwise.} \end{cases} \tag{5}$$

By the definition of the doubling algorithm, we have $|X_t| \leqslant n/16$ (otherwise we would have stopped), so there are at least $15n/16 \geqslant n/2$ vertices $v \in A_t$. The events that individual vertices are in B_t are independent (because we do not 're-test' edges). Hence if Z is a random variable with the $\mathrm{Bin}\big(n/2, q_{t,1}q_{t,2}\big)$ distribution, then $s_t \geqslant \mathbb{P}(Z \geqslant x_t)$.

From (5) it is easy to check that

$$\mathbb{E}[Z] = (n/2)q_{t,1}q_{t,2} \geqslant 2x_t.$$

Indeed, if $p_1 x_t \leqslant p_2 x_t < 2$, then $\mathbb{E}[Z] \geqslant (n/32)p_1 p_2 x_t^2 \geqslant 2x_t$ since $np_1 p_2 \geqslant c/(\log n)$ (from (2)) and $x_t \geqslant (\log n)^{3/2}$. If $p_1 x_t < 2 \leqslant p_2 x_t$, then $\mathbb{E}[Z] \geqslant np_1 x_t/16 \geqslant 2x_t$, using $np_1 \geqslant c\log n$. Finally, if $2 < p_1 x_t \leqslant p_2 x_t$, then $\mathbb{E}[Z] \geqslant n/8 \geqslant 2x_t$ since $x_t \leqslant n/16$. We thus have

$$s_t \geqslant \mathbb{P}(Z \geqslant x_t) \geqslant \mathbb{P}\big(Z \geqslant \mathbb{E}[Z]/2\big) \geqslant 1 - \exp\big(-\Omega(\mathbb{E}[Z])\big),$$

using a standard (Chernoff-type) bound for the final step. Now $q_{t,i}$ is increasing in t, so

$$\mathbb{E}[Z] = nq_{t,1}q_{t,2}/2 \geqslant nq_{0,1}q_{0,2}/2 \geqslant c(\log n)^2,$$

where the last inequality follows from (5), recalling that $x_0 = t_1 = (\log n)^{3/2}$ and that $p_1 \leqslant p_2 \leqslant (\log n)^{-2}$ (see (3)). This completes the proof. $\qquad \square$

Lemma 9 is now immediate.

Proof of Lemma 9. Recall that we wish to show that

$$\mathbb{P}\Big(I\big(G^{(1)} \cup G^{(2)}, n/16\big) \;\Big|\; I\big(G^{(1)}, (\log n)^{3/2}\big)\Big) \geqslant 1 - n^{-100}. \qquad (6)$$

Let t_2 be maximal such that $x_{t_2} \leqslant n/16$, which in particular implies that $t_2 = O(\log n)$. By Lemma 11, the left-hand side of (6) is at least

$$\prod_{t=0}^{t_2} \mathbb{P}\big(|B_t| \geqslant x_t \mid |X_t| = x_t\big) \geqslant \Big(1 - \exp\big(-\Omega(c(\log n)^2)\big)\Big)^{O(\log n)}$$

$$\geqslant 1 - \exp\big(-\Omega(c(\log n)^2)\big),$$

which is certainly at least $1 - n^{-100}$, as required. $\qquad \square$

3.3 Part III

It remains to show that if $G^{(1)} \cup G^{(2)}$ contains an internally spanned set X of size at least $n/16$ then $G = G^{(1)} \cup G^{(2)} \cup G^{(3)}$ is internally spanned with high probability. But this is trivial: using the final sprinkle, i.e., the edges of $G^{(3)}$, every vertex $v \in [n] \setminus X$ is joined to X by both a red edge and a blue edge, with high probability.

Proof of Theorem 2. As noted above, it remains only to prove (ii), and in doing so, we may assume (2). By Lemmas 4 (Part I) and 9 (Part II), $G^{(1)} \cup G^{(2)}$ contains an internally spanned set of size at least $n/16$ with probability at least $1 - 2n^{-100}$. Let X be such a set. Then the probability that there is any vertex of $[n] \setminus X$ not joined to X by at least one edge of $G_1^{(3)}$ (a red edge from the third sprinkling) and at least one edge of $G_2^{(3)}$ (a blue edge from the third sprinkling) is at most

$$2n(1 - p_1)^{n/16} = o(1),$$

since $p_1 \geqslant c(\log n)/n$ and c is sufficiently large. $\qquad \square$

References

[1] M. Aizenman and J.L. Lebowitz, *Metastability effects in bootstrap percolation*, J. Phys. A **21** (1988), no. 19, 3801–3813.

[2] C.D. Brummitt, S. Chatterjee, P.S. Dey, and D. Sivakoff, *Jigsaw percolation: What social networks can collaboratively solve a puzzle?*, Ann. Appl. Probab. **25**, (2015), no. 4, 2013–2038.

[3] S.V. Buldyrev, R. Parshani, G. Paul, H.E. Stanley, and S. Havlin, *Catastrophic cascade of failures in interdependent networks*, Nature **464** (2010), 1025–1028.

[4] R. Cerf and E.N.M. Cirillo, *Finite size scaling in three-dimensional bootstrap percolation*, Ann. Probab. **27** (1999), no. 4, 1837–1850.

[5] J. Gravner and D. Sivakoff, *Nucleation scaling in jigsaw percolation*, Ann. Appl. Probab. **27** (2017), no. 1, 395–438.

[6] A.E. Holroyd, *Sharp metastability threshold for two-dimensional bootstrap percolation*, Probab. Theory Related Fields **125** (2003), no. 2, 195–224.