

A Trust Model in Bootstrap Percolation

Rinni Bhansali and Laura P. Schaposnik

Abstract

Bootstrap percolation is a class of monotone cellular automata describing an activation process which follows certain activation rules. In particular, in the classical r -neighbor bootstrap process on a graph G , a set A of initially infected vertices spreads by infecting vertices with at least r already-infected neighbors. Motivated by the study of social networks and biological interactions through graphs, where vertices represent people and edges represent the relations amongst them, we introduce here a novel model which we name \mathcal{T} -bootstrap percolation (\mathcal{T} -BP). In this new model, vertices of the graph G are assigned random labels, and the set of initially infected vertices spreads by infecting (at each time step) vertices with at least a fixed number of already-infected neighbors of each label.

The Trust Model for Bootstrap Percolation allows one to impose a preset level of skepticism towards a rumor, as it requires a rumor to be validated by numerous groups in order for it to spread, hence imposing a predetermined level of trust needed for the rumor to spread. By considering different random and non-random networks, we describe various properties of this new model (e.g., the critical probability of infection and the confidence threshold), and compare it to other types of bootstrap percolation from the literature, such as \mathcal{U} -bootstrap percolation. Ultimately, we describe its implications when applied to rumor spread, fake news, and marketing strategies, along with potential future applications in modeling the spread of genetic diseases.

Contents

1	Introduction	4
1.1	Defining the \mathcal{T} -BP	4
1.2	Summary of main results	5
1.3	Bootstrap Percolation	5
1.3.1	Classical Bootstrap Percolation	5
1.3.2	\mathcal{U} -bootstrap percolation	6
1.4	Other Models	7
2	A Trust Model for Bootstrap Percolation	7
2.1	Immunity in \mathcal{T} -BP	7
2.1.1	Immunity for \mathcal{T} -BP as r -bootstrap percolation	8
2.1.2	Immunity for the \mathcal{T} -BP	10
2.1.3	A Network's Immunity	12
2.2	Immunity on Hierarchical Networks	12
2.3	Critical Probability	13
3	Two-Dimensional Trust Model for Bootstrap Percolation	13
3.1	Critical probability	13
3.2	\mathcal{T} -BP as \mathcal{U} -Bootstrap Percolation	14
3.2.1	Striped \mathcal{T} -BP	14
3.2.2	Diagonal BP	14
3.3	Other two dimensional \mathcal{T} -BP	15
3.3.1	3-1 BP	15
3.3.2	Checkerboard BP	16
3.3.3	Mixed BP	17
3.4	Trust Model for \mathcal{T} -BP for $m = 2$ as 2-bootstrap percolation on \mathbb{Z}^2	18
4	\mathcal{T}-BP on Random Networks	19
4.1	Varying Density	19
4.2	Varying the number of required labels	20
5	Conclusion	20
6	Acknowledgements	21
7	References	22

1 Introduction

Sooner or later most people struggle to find the perfect present: we hear from our son’s friends that certain remote-controlled helicopter would be so fun for his 10th birthday – but will it be safe? Once we hear from our son’s friends’ parents that the helicopter is indeed entertaining and safe for his age, we are closer to buying it. Is this recurrent phenomenon a consequence of a natural instinct that one has, where having the same information transmitted by different “types” of people makes it become more trustworthy? If so, we may wonder naturally:

How many different types of people (colleagues, friends, Uber drivers, etc.) should we hear a piece of news from, before we start transmitting it as a true fact?

Requiring a clear range for sources of information would allow members of a society to express skepticism towards gossip and fake news. Moreover, understanding this range could allow various industries to target wisely a minimum amount of consumers within each type of people, and use the natural spreading process of information to continue the marketing on its own.

Bootstrap percolation is a particular class of monotone cellular automata describing an activation process which follows certain activation rules, and which has been used to model interactions within societies. In particular, in the classical r -neighbor bootstrap process on a graph G , a set A of initially “infected” vertices spreads by infecting vertices with at least r already-infected neighbors. We shall introduce here a modified version of this model, which we call *Trust Model in Bootstrap Percolation* (\mathcal{T} -BP), designed to answer an equivalent question to those posed above:

How does information percolate when a messenger only passes the information on if it has been received from a number of different sources of certain types?

1.1 Defining the \mathcal{T} -BP

Consider a finite or infinite graph G , a number $m \in \mathbb{N}$, a *trust vector* $K = (k_1, \dots, k_m) \in \mathbb{N}^m$ of non-negative numbers, and a set $A := A_0$ of initially “infected” vertices in G . After assigning randomly a label in $\{1, \dots, m\}$ to each vertex, we define bootstrap percolation with trust vector K (this is, a trust model for bootstrap percolation with level K) as the process where at each time, all of the vertices which have at least k_i adjacent vertices infected with label i become infected. Hence, at each time t , the set of infected vertices is given by

$$A_{t+1} = A_t \cup \{v \in V(G) : |N_i(v) \cap A_t| \geq k_i \text{ for } i = 1, \dots, m\},$$

where $N_i(v)$ denotes the set of adjacent vertices to v in G with label i .

Yet while it is of interest to study bootstrap percolation for a fixed trust vector K , it is also important to consider models where families of trust vectors are allowed at the same time. Following the idea of an update family \mathcal{U} from [8], as described in Section 1.3.2, we shall consider a *trust family* $\mathcal{T} := \{K^1, \dots, K^n\}$ of trust

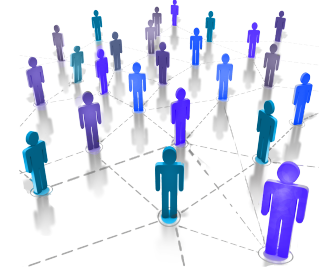


Figure 1: Examples of a social network where different colors–labels–describe the different types of people needed for gossip spreading.

vectors $K^j = (k_1^j, \dots, k_m^j) \in \mathbb{N}^m$. Then, we shall refer to **Trusted Bootstrap Percolation**, or \mathcal{T} -**bootstrap percolation** (\mathcal{T} -BP), as the percolation process where $A_0 := A$ and

$$A_{t+1} = A_t \cup \{v \in V(G) : \exists K^j \in \mathcal{T} \text{ s.t. } |N_i(v) \cap A_t| \geq k_i^j \text{ for } i = 1, \dots, m\}. \quad (1)$$

$$A_{t+1} = A_t \cup \{v \in V(G) : \exists K^j \in \mathcal{T} \text{ s.t. } |N_i(v) \cap A_t| \geq k_i^j \text{ for } i = 1, \dots, m\}. \quad (2)$$

Additionally, denote by $f : V(G) \rightarrow 1, \dots, m$ the function that assigns the random labels to each vertex of G and note that $N(V) = \bigcup_{i=1}^m N_i(V)$.

1.2 Summary of main results

Along the paper we study the above new model, \mathcal{T} -bootstrap percolation, from different perspectives. First, in Section 2, we look at immunity of vertices, where a vertex v is *immune* if for $v \notin A_0$, then $v \notin A_\infty$. We compute the probability that a vertex v of degree d is immune for the following models:

- **\mathcal{T} -BP as r -bootstrap percolation (see Proposition 7).** Consider \mathcal{T} -BP on a graph G with multiple trust vectors of length m whose entries are $k_j \in \{0, 1\}$, with exactly r non-zero.
- **\mathcal{T} -BP with single trust vector K (see Proposition 14).** Consider \mathcal{T} -BP on a graph G with a single trust vector K with entries (k_1, k_2, \dots, k_m) .

Inspired by the regular structure of natural systems in chemistry and biology, in Section 3 we study the \mathcal{T} -bootstrap percolation on the following two dimensional regular networks which can or can not be understood through \mathcal{U} -bootstrap percolation:

- **Striped lattice** (A case of \mathcal{U} -bootstrap percolation, see Section 3.2.1)
- **Diagonal lattice** (A case of \mathcal{U} -bootstrap percolation, see Section 3.2.2)
- **Three-to-One lattice** (A case of non- \mathcal{U} -bootstrap percolation, see Section 3.3.1)
- **Checkerboard lattice** (A case of non- \mathcal{U} -bootstrap percolation, see Section 3.3.2)
- **Mixed lattice** (A case of non- \mathcal{U} -bootstrap percolation, see Section 3.3.3)

Finally, since social networks are best modeled by random graphs with certain particular density properties, we dedicate Section 4 to the numerical study of the \mathcal{T} -BP on random networks.

1.3 Bootstrap Percolation

Bootstrap percolation was first introduced in the context of disordered magnetic systems in 1979 by Chalupa, Leath and Reich [10], where the classical r -neighbor bootstrap process on a graph G was studied. Since then, this process has been investigated on several different kinds of networks including finite grids [1, 3, 12, 13], trees [4], and random networks [5, 14], and with many different applications in fields such as sociology [11, 20], physics [17], and epidemiology [7].

1.3.1 Classical Bootstrap Percolation

It is important to note that classical r -neighbor bootstrap percolation is a particular case of \mathcal{T} -bootstrap percolation. In particular, one may consider $m = 1$, and set $\mathcal{T} = \{K^1 = (r)\}$ to recover r -neighbor bootstrap percolation with only one label, in which if an uninfected vertex has r infected neighbors, it will become infected in the next time step.

1.3.2 \mathcal{U} -bootstrap percolation

Recently Bollobás, Smith and Uzzell introduced in [8] a class of percolation models called \mathcal{U} -bootstrap percolation, which includes r -neighbor bootstrap percolation as a special case. These models have certain update rules encoded in a set \mathcal{U} , and in the case of two dimensional lattices, they divided the model into three classes: subcritical, critical and supercritical. In what follows, we shall describe these models and in the following sections, we shall study different types of \mathcal{T} -BP and describe when they can be obtained as particular cases of \mathcal{U} -bootstrap percolation.

As in the various bootstrap percolation models, consider a graph G with vertex set $V(G)$. For \mathcal{U} -bootstrap percolation in particular, let $G = \mathbb{Z}^2$. Then, consider a set $A_0 \subset V(G)$ of initially infected vertices, where each vertex $v \in V(G)$ has probability p of being in A_0 . In order to determine how this initial infection spreads, consider a finite collection of finite, non-empty subsets $U = \{X_1, X_2, \dots, X_n\}$ of G . Given the above, in a \mathcal{U} -bootstrap percolation process on G , the set A_t of infected vertices at time t is defined as $A_{t+1} = A_t \cup \{x \in G \mid \text{there exists } i \in [n] \text{ such that } X_i + x \in A_t\}$, where $A_0 := A$ is the set of initially infected vertices, the set \mathcal{U} is referred to as the *update family*, and each X_i as an *update rule*.

Remark 1. One can see that r -neighbor bootstrap percolation on \mathbb{Z}^d is an example of a \mathcal{U} -bootstrap percolation model, where the update rules are the $\binom{2d}{r}$ r -subsets of the neighbors of the origin.

Along the paper we shall write $[A]$ for the set of all vertices that eventually become infected $[A] = \bigcup_{t=0}^{\infty} A_t$, and say that A is closed under \mathcal{U} if we have $[A] = A$. Following the notation of [8], we consider the unit circle $S^1 \in \mathbb{R}^2$, and for each unit vector $u \in S^1$, let \mathbb{H}_u denote the discrete half plane $\mathbb{H}_u = \{x \in \mathbb{Z}^2 : \langle x, u \rangle < 0\}$.

Definition 2. A *stable direction* for \mathcal{U} is a unit vector u such that $[\mathbb{H}_u] = \mathbb{H}_u$, or in other words, one such that no new sites become infected when the initial set is equal to the half plane \mathbb{H}_u . Otherwise, u is said to be an *unstable direction* for \mathcal{U} . The set \mathcal{S} for \mathcal{U} is the *set of all stable directions* for the update family \mathcal{U} :

$$\mathcal{S} := S(\mathcal{U}) = \{u \in S^1 \mid u \text{ is stable for } \mathcal{U}\}.$$

An update rule X is said to *destabilize* a direction $u \in S^1$ if for $\mathcal{U} = \{X\}$ one has that $u \notin S(\mathcal{U})$. As mentioned earlier, it is shown in [8] that \mathcal{U} -bootstrap percolation models can be classified into three types, depending on their update families \mathcal{U} :

- **Supercritical:** When there exists an open semicircle $C \subset S^1$ which is disjoint from the set of stable directions $\mathcal{S} \cap C = \emptyset$.
- **Critical:** When $\mathcal{S} \cap C \neq \emptyset$ for every open semicircle $C \subset S^1$, and there is an open semicircle $C \subset S^1$ that is disjoint from $\text{Int}(\mathcal{S})$.
- **Subcritical:** When every open semicircle in S^1 has non-empty intersection with $\text{Int}(\mathcal{S})$.

When studying percolation on a graph, one of the most important quantities to consider is the critical probability p_c of the model: this is the probability of initial infection at which the probability of percolation reaches $\frac{1}{2}$, or the threshold point. Note that percolation means that the entire graph G is infected at $t = \infty$, which is equivalent to saying $[A_0] = A_\infty = V(G)$. Notably, through the above classification system, one can obtain insight into the value of the critical probability of infection solely based on the update family \mathcal{U} .

Definition 3. The *critical probability of infection* is defined as

$$p_c = \inf \left\{ p \mid \mathbb{P}_p(A_\infty = V(G)) \geq \frac{1}{2} \right\}. \quad (3)$$

Remark 4. In infinite graphs, $\mathbb{P}_p(A_\infty = V(G))$ is always 0 or 1 due to ergodicity (as percolation is a translation invariant) [8]. In particular, this is the setting considered in [8], and thus in such cases $p_c = \inf \{p \mid \mathbb{P}_p(A_\infty = V(G)) = 1\}$.

When considering critical and supercritical models, it is proven in [8] that $p_c = 0$, whilst in the case of subcritical models, it is proven in [2] that $p_c > 0$. In later sections, we shall consider the methods of [2] to develop similar notation and concepts in order to address the numerous similarities between specific forms of the Trust Model in Bootstrap Percolation and subcritical \mathcal{U} -bootstrap percolation.

1.4 Other Models

However, while the activation process for the \mathcal{T} -BP is primarily inspired by bootstrap percolation models, we had many other inspirations as well: One example is the work done in [16], which uses graphs to represent society and considers various constraints to inhibit the spread of a rumor. In their work, the authors only allow the rumor to spread to vertices which would be “interested” in it. In a similar fashion, our \mathcal{T} -BP allows a rumor to spread only to vertices which are able to deem it “unbiased”. Additionally, the use of hierarchical graphs of [16] inspired our use of those networks in Section 2.2.

From a different perspective, [9] introduces the existence of trust (and distrust) among members of a society. However, this sense of skepticism is introduced differently in [9] than in the \mathcal{T} -BP. Their model bases trust on how much one trusts each person telling one the rumor, whereas the \mathcal{T} -BP bases trust on how many different types of people tell one the rumor (ignoring the *strength* of one’s relationships). This difference is crucial because their model emphasizes psychological phenomena such as group-think and confirmation bias, as one believe rumors which one’s “friends” (and not one’s foes) tell one, whereas ours is meant to *combat* these phenomena.

2 A Trust Model for Bootstrap Percolation

Returning to the question from the introduction, we ask: *How does information percolate when a messenger only passes the information if it has been received by a number of different sources of certain types?* Motivated by this, we begin our study of the *Trust Model for Bootstrap Percolation*, defined in Section 1.1.

2.1 Immunity in \mathcal{T} -BP

In the following subsections we shall focus on two forms of \mathcal{T} -bootstrap percolation that are of particular interest: \mathcal{T} -BP as r -bootstrap percolation, and the simplest \mathcal{T} -BP, which has single trust vector K . It should be noted that within the \mathcal{T} -BP, a proportion of the vertices will be *immune* to the disease: these vertices cannot belong to A_t for any $t \in \mathbb{N}$ *unless* they are in A_0 , as they cannot become infected from their neighbors. We shall formally define the set of immune vertices by

$$\mathcal{I} := \left\{ v \in V(G) : \forall K^j \in \mathcal{T}, \exists i \in [m] \text{ s.t. } |N_i(v)| < k_i^j \right\} \quad (4)$$

where $\mathcal{T} = \{K^1, \dots, K^n\}$ and $K^j = \{k_1^j, \dots, k_m^j\} \in \mathbb{N}^m$ as before.

In order to study the \mathcal{T} -BP on G , the vertices in \mathcal{I} need to be removed from G , leading to a modified graph which sometimes will be disconnected, with some components that might never become infected depending on the set A_0 . We shall denote by $\mathcal{G} := G - \mathcal{I}$ the graph where all immune vertices and corresponding edges have been removed.

In order to understand the likelihood of percolation of a \mathcal{T} -BP model and how immune vertices disrupt percolation on a network, it is useful to introduce the notion of *diversity*:

Definition 5. The *diversity* of a vertex $v \in G$ in a \mathcal{T} -BP model is the number D_v of different labels that vertices in $N(v)$ have:

$$D_v := |\{i \in \{1, \dots, m\} : N_i(v) \neq \emptyset\}|. \quad (5)$$

2.1.1 Immunity for \mathcal{T} -BP as r -bootstrap percolation

In what follows we shall study immunity of vertices for a form of \mathcal{T} -BP known as “ \mathcal{T} -BP as r -bootstrap percolation.” The \mathcal{T} -BP as r -bootstrap percolation model has trust vectors $K^i \in \mathbb{N}^m$ which have exactly r non-zero entries given by 1. In this setting, the cardinality of the trust family $|\mathcal{T}|$ is $\binom{m}{r}$, and in this model, at least r *different* neighbors must be infected in order for a vertex to be infected (“different neighbors” meaning neighboring vertices with distinct labels).

However, even before the infection is introduced to a \mathcal{T} -bootstrap percolation network, there exist certain vertices which will never be able to contract the infection (unless, of course, they are infected initially): these vertices are defined as *immune*, and we shall describe them as follows: In the case of \mathcal{T} -BP as r -bootstrap percolation, a vertex is *immune* if it does not have neighbors of at least r distinct labels.

Example 6. Consider \mathcal{T} -BP as 3-bootstrap percolation. In this case, the model can be used to represent the spread of a political rumor among a society of Democrats, Republicans, and Independents. Then, labeling the vertices with 1, 2, 3 to represent each political party, suppose that, in order to limit the spread of biased (and potentially false) information, there exists a rule that an individual will only believe and pass on the rumor, if he/she heard it from 2 people with different political backgrounds. Then, the trust family in this model would be $\mathcal{T} = \{(1, 1, 0), (1, 0, 1), (0, 1, 1)\}$, and this is equivalent to 2-neighbor bootstrap percolation with 3 labels. Yet, if an individual knows only Democrats, then it is impossible for them to get infected. Thus, they are immune.

From the definition of \mathcal{T} -BP as r -bootstrap percolation, one can see that if a vertex is not immune, then $D_v \geq r$. Hence, it is of particular interest to understand which vertices v have $D_v < r$, since those will comprise all of the immune vertices.

Proposition 7. Consider \mathcal{T} -BP on a graph G with multiple trust vectors of length m whose entries are $k_j \in \{0, 1\}$, with r non-zero. Then, for $\{j^d\}$ the Stirling number of the second kind, the probability of immunity $p_0(d, m, r)$ for a vertex v with $|N(v)| = d$ (equivalent to the probability that $D_v \leq r - 1$) is

$$p_0(d, m, r) := P(D_v \leq r - 1) = \frac{\sum_{j=1}^{r-1} \left[\binom{m}{j} (j!) \{j^d\} \right]}{m^d}. \quad (6)$$

Proof. We shall prove (6) through the principle of inclusion and exclusion. In order to do this, we shall first calculate the probabilities $P(D_v = i)$ for $i = 1, \dots, r$, and use this to calculate $p_0(d, m, r)$.

Given a fixed integer n such that $1 \leq n \leq r - 1$, in order to understand $P(D_v = n)$ note that there are $\binom{m}{n}$ ways to choose the n acceptable labels that the neighbors may have, or in other words, ways to choose a set $N \subset \{1, \dots, m\}$ with $|N| = n$. Once these labels are chosen, the number of surjective functions (1.1) such that $f(N(v)) = N$ is given by

$$\sum_{i=0}^n (-1)^{n-i} \binom{n}{i} (n-i)^d. \quad (7)$$

To see this, we begin by noting that each of the d adjacent vertices in $N(v)$ has n possible values for its label, so there are n^d functions from $N(v)$ to N . However, not all functions will be different. To visualize this, consider a Venn diagram where each circle A_s of the Venn diagram corresponds to a label s , with $1 \leq s \leq m$, and is defined as $A_s := \{f : N(v) \rightarrow N : s \notin N\}$. For instance, every object in A_1 will be a possible configuration of neighbors of v such that none of them are of label 1.

All functions which are not surjective to N must appear in some set A_i for $i \in N$. Therefore, the number of onto functions $f : N(V) \rightarrow N$ is given by $n^d - |\bigcup_{i \in N} A_i|$. To calculate this, use the Principle of Inclusion and Exclusion [18]. Thus, to find $|\bigcup_{i \in N} A_i|$, consider $\sum_{i \in N} |A_{x_1} \cap \dots \cap A_{x_j}|$ for all j such that $1 \leq j \leq n$, and $x_i \in N$, which is given by

$$\sum |A_{x_1} \cap \dots \cap A_{x_j}| = \binom{n}{j} (n-j)^d, \quad (8)$$

since there are $\binom{n}{j}$ ways to choose the j labels out of N that the neighbors cannot occupy, and so there are $n-j$ options for every neighbor's label. Thus the number of surjective functions $f : N(v) \rightarrow N$ is

$$\sum_{i=0}^n (-1)^{n-i+1} \binom{n}{i} (n-i)^d. \quad (9)$$

From the above, one can calculate the number of different label assignments that the vertices $N(v)$ can have, where $D_v = j$: there are $\binom{m}{j}$ possibilities for j labels, which multiplied by (9) leads to

$$\binom{m}{j} \sum_{i=0}^j (-1)^{j-i+1} \binom{j}{i} (j-i)^d. \quad (10)$$

Equivalently, we can write this as $\binom{m}{j} (j!) \{j^d\}$ where $\{j^d\}$ is the Stirling number of the second kind [19]. Hence, the number of different label assignments to the graph G for which the diversity $D_v \leq r - 1$ is

$$\sum_{j=1}^{r-1} \left[\binom{m}{j} (j!) \{j^d\} \right]. \quad (11)$$

Recalling that there are m^d possible functions $f : N(v) \rightarrow \{1, \dots, m\}$, the probability of a vertex having diversity $D_v \leq r - 1$ is given by

$$P(D_v \leq r - 1) = \frac{\sum_{j=1}^{r-1} \left[\binom{m}{j} (j!) \{j^d\} \right]}{m^d}, \quad (12)$$

which concludes the proof. \square

Corollary 8. *In the setting of Proposition 7, the number of vertices of degree d in \mathcal{I} satisfies*

$$|\mathcal{I}| \geq \sum_{j=1}^{r-1} \left[\binom{m}{j} (j!) \left\{ \begin{matrix} d \\ j \end{matrix} \right\} \right]. \quad (13)$$

Whilst in coming sections we shall analyze the behavior of the \mathcal{I} -BP on different graphs, the case of lattices in \mathbb{Z}^d will be of much interest, since these will illustrate some general properties of the model.

2.1.2 Immunity for the \mathcal{I} -BP

The second form of \mathcal{I} -bootstrap percolation we shall consider is the simplest form of \mathcal{I} -BP in which the infection grows with a single rule $K = (k_1, k_2, \dots, k_m)$. In this setting, given any vertex v with degree d on a graph G , consider the vector $x = (x_1, x_2, \dots, x_m)$ defined by $x_i = |N_i(v)|$, where in particular, $\sum_{i=1}^m x_i = d$. In order for a vertex to be immune under in this setting, one must have $x_i < k_i$ for some integer i such that $1 \leq i \leq m$. We will rely on this fact for the following proposition:

Proposition 9. *Consider the \mathcal{I} -BP, with $K = (k_1, k_2, \dots, k_m)$. Then, the probability of immunity $p_0(d, K)$ for a vertex v with $|N(v)| = d$ is*

$$p_0(d, K) = 1 - \sum_{x_m=k_m}^{d-(\sum_{l=1}^{m-1} k_l)} \binom{d}{x_m} \left(\frac{1}{m}\right)^{x_m} \left[\sum_{x_{m-1}=k_{m-1}}^{d-x_m-(\sum_{l=1}^{m-1} k_l)} \binom{d-x_m}{x_{m-1}} \left(\frac{1}{m}\right)^{x_{m-1}} \left[\dots \right. \right. \\ \left. \left. \left[\sum_{x_2=k_2}^{d-(\sum_{l=3}^m x_l)-k_1} \binom{d-(\sum_{l=3}^m x_l)}{x_2} \left(\frac{1}{m}\right)^{x_2} \left(\frac{1}{m}\right)^{d-(\sum_{l=2}^m x_l)} \right] \dots \right] \right].$$

Proof. Let $f(a, b, c)$ be the function defined as

$$f(a, b, c) := \sum_{x_i=b_i}^{a-(\sum_{l=1}^{i-1} b_l)} \binom{a}{x_i} \left(\frac{1}{c}\right)^{x_i} \left[\sum_{x_{i-1}=b_{i-1}}^{a-x_i-(\sum_{l=1}^{i-1} k_l)} \binom{a-x_i}{x_{i-1}} \left(\frac{1}{c}\right)^{x_{i-1}} \dots \right. \\ \left. \left[\sum_{x_2=b_2}^{a-(\sum_{l=3}^i x_l)-b_1} \binom{a-(x_i+\dots+x_3)}{x_2} \left(\frac{1}{c}\right)^{x_2} \left(\frac{1}{c}\right)^{a-(\sum_{l=2}^i x_l)} \right] \dots \right]$$

where $b = \{b_1, b_2, \dots, b_i\}$ and $a, c \in \mathbb{N}$. Then, our claim can equivalently be stated as $p_0 = 1 - f(d, k, m)$ for $k = \{k_1, k_2, \dots, k_m\}$. Indeed, this can be proven with an inductive argument on the number of available labels m .

For $m = 2$, the vector $k = \{k_1, k_2\}$ satisfies $k_1 + k_2 \leq d$. We must find the probability of assigning d vertices to one of 2 labels such that there are at least k_1 vertices of label 1 and k_2 of label 2—this holds when a vertex is not immune, so we must then subtract this from 1. This is equivalent to saying there may be x_1 vertices of label 1 such that $k_1 \leq x_1 \leq d - k_2$, and all other vertices of label 2. Note that the probability that there are x_1 vertices of label 1 is $\binom{d}{x_1} \left(\frac{1}{2}\right)^{x_1} \left(\frac{1}{2}\right)^{d-x_1}$.

By summing over all possible values for x_1 varying from k_1 to $d - k_2$, we obtain the overall p_0 for $m = 2$ and $K = \{k_1, k_2\}$ to be $p_0(d, \{k_1, k_2\}) = 1 - f(d, \{k_1, k_2\}, 2) = 1 - \sum_{x_1=k_1}^{d-k_2} \binom{d}{x_1} \left(\frac{1}{2}\right)^{x_1} \left(\frac{1}{2}\right)^{d-x_1}$. Now, move on to $m = 3$ with $k = \{k_1, k_2, k_3\}$. We must find the probability of assigning d vertices to one of 3 labels such that there are at least k_1 vertices of label 1, k_2 of label 2, and k_3 vertices of label 3. We can approach this with casework:

First, note that there may be x_1 vertices of label 1 such that $k_1 \leq x_1 \leq d - (k_2 + k_3)$. Then, take cases based on the value of x_1 . Note that given x_1 , there are $d - x_1$ remaining vertices to consider, with 2 possible labels to assign them to. Now this question is almost the same as the one with $m = 2$, with the only difference being that the probability a vertex occupies one of these two labels (2 or 3) is not $\frac{1}{2}$, but $\frac{1}{3}$. So given x_1 , the probability that the number of vertices of label 2 is greater than k_2 and the number of vertices of label 3 is greater than k_3 is just $f(d - x_1, \{k_2, k_3\}, 3)$. This means that given x_1 , the probability that the vertex is initially not immune (we are using complementary counting) is:

$$\binom{d}{x_1} \left(\frac{1}{3}\right)^{x_1} f(d - x_1, \{k_2, k_3\}, 3). \quad (14)$$

Now, as x_1 can go from k_1 to $d - (k_2 + k_3)$, p_0 for $m = 3$ is:

$$p_0(d, \{k_1, k_2, k_3\}) = 1 - \sum_{x_1=k_1}^{d-(k_2+k_3)} \binom{d}{x_1} \left(\frac{1}{3}\right)^{x_1} f(d - x_1, \{k_2, k_3\}, 3) \quad (15)$$

$$= 1 - f(d, \{k_1, k_2, k_3\}, 3), \quad (16)$$

which provides the intuition for the inductive step: we must prove that

$$p_0(d, \{k_1, \dots, k_m\}) = 1 - f(d, \{k_1, \dots, k_m\}, m),$$

assuming that there exists an i such that $p_0(d, \{k_1, \dots, k_i\}) = 1 - f(d, \{k_1, \dots, k_i\}, i)$.

In order to prove the statement by induction, consider $m = i + 1$, and let $k = \{k_1, k_2, \dots, k_{i+1}\}$. Using casework once more, let the number of vertices of label 1 be $x_1 \in \{k_1, \dots, d - \sum_{j=2}^{i+1} k_j\}$.

Then, there are $d - x_1$ vertices which must have labels in $[i + 1]/\{1\}$. By the inductive assumption, the probability that they have at least k_j vertices of label j is $f(d, \{k_2, \dots, k_{i+1}\}, i)$. However, note that we must replace the $\frac{1}{i}$ terms in this formula with $\frac{1}{i+1}$ because the probability any individual vertex has a specific label is now $\frac{1}{i+1}$, meaning that it is actually $f(d, \{k_2, \dots, k_{i+1}\}, i + 1)$. Also, the probability that there are x_1 vertices of label 1 is $\binom{d}{x_1} \left(\frac{1}{i+1}\right)^{x_1}$, so multiplying these two terms one finds that the probability that a vertex is initially *not* immune given that it has x_1 neighbors of label 1 is :

$$\binom{d}{x_1} \left(\frac{1}{i+1}\right)^{x_1} f(d, \{k_2, \dots, k_{i+1}\}, i + 1). \quad (17)$$

Since x_1 can be anything from k_1 to $d - \sum_{j=2}^{i+1} k_j$, summing over the possible values of x_1 and subtracting this summation from 1 leads to the probability

$$p_0(d, \{k_1, k_2, \dots, k_{i+1}\}) = 1 - \sum_{x_1=k_1}^{d-(\sum_{j=2}^{i+1} k_j)} \binom{d}{x_1} \left(\frac{1}{i+1}\right)^{x_1} f(d, \{k_2, \dots, k_{i+1}\}, i + 1) \quad (18)$$

$$= 1 - f(d, \{k_1, \dots, k_{i+1}\}, i + 1). \quad (19)$$

This concludes the inductive step, finalizing the proof. \square

2.1.3 A Network's Immunity

The formulas for p_0 from the previous sections relate only to single vertices. In order to apply this concept to entire graphs, we look at the expected fraction of immune vertices on any given graph, as a function of its degree distribution $P(d)$, which is defined to be the fraction of nodes in the graph with degree d .

Definition 10. We shall denote by $p_0(G, \mathcal{T})$ the *expected fraction of immune vertices* for a \mathcal{T} -BP on a graph G , and by $p_0(d, \mathcal{T})$ the *probability of immunity* for a vertex of degree d in G .

Corollary 11. *In a \mathcal{T} -BP on a graph G with degree distribution $P(d)$, one has that the expected fraction of immune vertices is*

$$p_0(G, \mathcal{T}) = \sum_{j=0}^{\infty} p_0(j, \mathcal{T})P(j). \quad (20)$$

2.2 Immunity on Hierarchical Networks

After finding the general formula for $p_0(G, \mathcal{T})$ in the previous section, we focus our study to hierarchical networks. We wish for the graphs we apply the \mathcal{T} -BP on to represent society. A notable quality

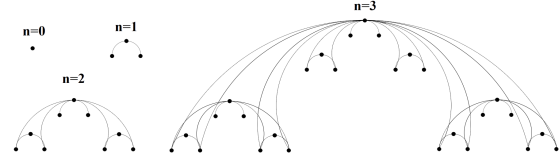


Figure 2: The network being built at each step n .

of many social networks is that they have a power law degree distribution, which means that the fraction $P(d)$ of vertices in G having degree d is approximately $P(d) \sim d^{-\gamma}$ for some constant $\gamma \in \mathbb{R}$. For social networks particularly, γ is often around between 2 and 3. However, while social networks are relatively random, graphs known as deterministic hierarchical networks have power law degree distributions while also having predetermined configurations, making them “simpler” to study. Thus, it would be interesting to investigate the \mathcal{T} -BP on a deterministic hierarchical network with γ being approximately 2 to 3.

One such example is the graph in Figure 2, defined in [6]: Start with one node, a root node. Add two more nodes and connect them to the root. At step n , add 3^{n-1} nodes each, identical to the figure in the previous iteration (step $n-1$) and connect the 2^n bottom nodes to the root. The degree exponent is $\gamma = 1 + \frac{\ln 3}{\ln 2}$. From this, and using results from [6], we discover the following proofs:

Lemma 12. *At n iterations, there are $2(3^{n-\log_2(d+2)})$ vertices of degree d .*

Proof. From [6], at n iterations, there are $(\frac{2}{3})3^{n-i}$ vertices with degree $2^{i+1} - 2$. Then, by substituting d with $2^{i+1} - 2$ and solving, we find that there are $2(3^{n-\log_2(d+2)})$ vertices of degree d at this step. \square

Corollary 13. *The probability a vertex has degree d is $2(d+2)^{-\log_2 3}$.*

Proof. There are also 3^n vertices total at n iterations. So the probability a vertex has degree d at n iterations is $\frac{2(3^{n-\log_2(d+2)})}{3^n}$, or $2(3^{-\log_2(d+2)})$, which can be simplified to $2(d+2)^{-\log_2 3}$. \square

Proposition 14. *The expected fraction of immune vertices $p_0(G, \mathcal{T})$ for this hierarchical network is*

$$\sum_{d=1}^{2^{n+1}-2} 2(d+2)^{-\log_2 3} p_0(j, \mathcal{T}).$$

Proof. According to the paper, the maximum degree of any vertex on this graph is the degree of the root, which at step n is $2^{n+1} - 2$. Hence, this is the upper bound of the summation. To find p_0 , or the expected fraction of initially immune vertices on this model with update vector set L , we need to find the probability of immunity for a vertex with j neighbors and then multiply that by the probability that a vertex has j neighbors. Finally, we need to sum this product across all possible j . Using this method, we find that

$$p_0(G, \mathcal{F}) = \sum_{d=1}^{2^{n+1}-2} 2(d+2)^{-\log_2 3} p_0(j, \mathcal{F}).$$

2.3 Critical Probability □

Having studied the immune vertices of the graph G in the previous section, we shall consider now a \mathcal{F} -BP for which the set A of initially infected vertices is chosen randomly from G , by setting the probability of being initially infected to be p . To understand this model, consider the critical probability p_c of infection for which the probability of percolation is $\frac{1}{2}$.

Proposition 15. *In a \mathcal{F} -BP on a graph G , the critical probability of infection satisfies $p_c \geq \frac{2p_0-1}{2p_0}$.*

Proof. The probability that there exists an immune vertex that is not initially infected is $p_0(1-p_c)$. If $p_0(1-p_c) > \frac{1}{2}$, then the probability of percolation is already less than $\frac{1}{2}$, as the existence of an uninfected dead vertex prevents percolation. So, a lower bound for the critical probability p_c is $\frac{2p_0-1}{2p_0}$. □

3 Two-Dimensional Trust Model for Bootstrap Percolation

Following the more general results, the primary focus of our paper lies on the \mathcal{F} -BP for $m = 2$ as 2-bootstrap percolation on \mathbb{Z}^2 . This model has $K = ((1, 1))$. The primary motivations for using the graph \mathbb{Z}^2 are because new bootstrap percolation models are often first studied on the 2D lattice and also because many of our results are inspired by the past study of \mathcal{U} -bootstrap percolation, which has only been studied on \mathbb{Z}^2 .

3.1 Critical probability

Recall the definition of p_c , the critical probability of infection. In [8], the authors prove that $p_c = 0$ for critical and super-critical models. The basis of their argument is that the lack of stable directions in these models allows for almost unrestrained growth of the infection from multiple directions, ultimately enabling percolation. In contrast, the argument made in [2] proves that $p_c > 0$ when the update family, \mathcal{U} , is subcritical. As a very general outline of the proof, the idea is that since so many directions are strongly stable-in particular, there are 3 such that the triangle formed by these 3 directions contains the origin-all growth will be bounded by triangles with side lengths perpendicular to these strongly stable directions if p is small enough, thus preventing percolation.

Our ultimate goal would be to prove the existence of similar “triangles” on \mathcal{F} -BP for $m = 2$ as 2-bootstrap percolation, and we conjecture that, similarly to subcritical \mathcal{U} -bootstrap percolation,

Conjecture 16. *For 2-bootstrap percolation with $m = 2$ on \mathbb{Z}^2 , $p_c > 0$.*

Thus, in the following sections, although we will study various specific models of 2-bootstrap percolation with $m = 2$ and derive results for each of these models, the motivation will be to develop a greater intuition for the general 2-bootstrap percolation model, along with terminology which will hopefully be useful in approaching our conjecture.

3.2 \mathcal{T} -BP as \mathcal{U} -Bootstrap Percolation

We shall dedicate this section to the study of a network on which \mathcal{T} -BP appears as a form of \mathcal{U} -bootstrap percolation.

3.2.1 Striped \mathcal{T} -BP

Begin with \mathbb{Z}^2 and assign to the vertices such that the graph appears “striped”. More rigorously, assign all vertices with even x -coordinate to label 0, and the remaining vertices to label 1. Now, the rules of \mathcal{T} -BP for $m = 2$ as 2-bootstrap percolation still apply: if a vertex has 2 of its neighbors with *different* labels infected, then it too becomes infected. So, note that for any vertex on this graph, its vertical neighbors (the neighbors above and beneath it on the lattice) have the same label, while its horizontal neighbors have the other label. This gives a fixed set of update rules for all the vertices, defining \mathcal{U} as follows:

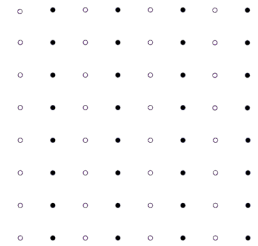


Figure 3: The lattice configuration for striped BP, with vertices of label 1 (black) and 2 (white).

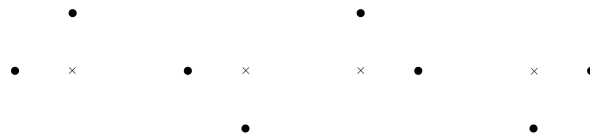


Figure 4: The update family \mathcal{U} for striped BP. There are 4 rules, shown here.

Lemma 17. *The update family \mathcal{U} for striped bootstrap percolation is critical.*

Proof. The first step is to find the stable directions. We claim that they are $\{\pm 1, 0\}$ and $\{0, \pm 1\}$. It is easy to verify that these work. Additionally, these four stable directions are enough to fulfill the first criteria: that there must exist a stable direction in every open semicircle $C \subset S^1$. Now, we must show that all other $u \in S^1$ are not stable. Let us first focus on vectors in the first quadrant. The proof can be extended to the other quadrants by symmetry. So, in the first quadrant, the border of any \mathbb{H}_u will have the following components:

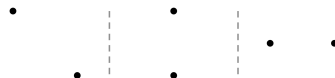


Figure 5: The configurations seen on the border of \mathbb{H}_u for unit vectors u in the first quadrant.

Having the first, diagonal, component defined in Figure 5 at some point on the boundary will result in a newly infected site from the fourth rule in \mathcal{U} . Therefore, all u in the first quadrant are unstable. And by the symmetry of the model, it is clear that the same holds for every quadrant—all u not on the axes will have the “diagonal components”, and therefore will be unstable. Thus, the only stable directions are the axes. Additionally, this model lacks any strongly stable directions, making it critical. \square

3.2.2 Diagonal BP

The next model is another form of 2-bootstrap percolation with $m = 2$, which again happens to be a form of \mathcal{U} -bootstrap percolation. This we call diagonal bootstrap percolation.

Begin with \mathbb{Z}^2 and assign labels 1 and 2 to the vertices such that the graph appears “diagonally striped”. Essentially, consider lines of the form $y = x + b$, for $b \in \mathbb{Z}$. If $b \equiv 0, 1 \pmod{4}$, make all vertices on the line have label 1, and otherwise, assign them label 2. Next, note that for any vertex on this graph, its neighbors up and to the left have the same label, while its neighbors down and to the right have the other label. This gives a fixed set of update rules for every vertex, which defines \mathcal{U} for this model:

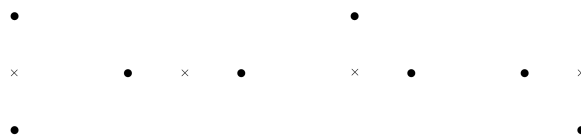


Figure 7: The update family \mathcal{U} for diagonal BP. There are 4 rules, shown here.

Lemma 18. *The update family \mathcal{U} for diagonal bootstrap percolation is subcritical.*

Proof. A subcritical update family is one such that for every open semicircle $C \subset S^1$, there is a strongly stable direction u such that $u \in C$. Now, we claim that every u in quadrants 2 and 4 are strongly stable under this update family \mathcal{U} —which would then satisfy the claim that it is subcritical.

First, it is clear that the axes are stable directions. Next, look at all unit vectors u in the second quadrant. Note that any such u will create an \mathbb{H}_u with the components similar to those Figure 5 somewhere on the border. The only difference is that the first component (the diagonal one), is reflected across the y -axis. Clearly, then, no u in the second quadrant will result in new vertices being infected, because X_3 and X_4 both will not destabilize any u in the second quadrant. A similar argument can be applied to the fourth quadrant, and ultimately we find that all u in quadrants 2 and 4 are stable, making \mathcal{U} subcritical. \square

3.3 Other two dimensional \mathcal{T} -BP

In what follows we shall introduce two different networks on which the \mathcal{T} -BP does *not* appear as a form of \mathcal{U} -bootstrap percolation.

3.3.1 3-1 BP

Next, we investigate a different form of 2-bootstrap percolation with $m = 2$ which is *not* a form of \mathcal{U} -bootstrap percolation. This we call 3-1 percolation, and because it is our first example of a model that is distinctly part of \mathcal{T} -BP, but not \mathcal{U} -bootstrap percolation, we use it to define terminology useful to describing \mathcal{T} -BP overall. Begin with \mathbb{Z}^2 and assign labels to the vertices such that the graph appears as shown in Figure 8.

Lemma 19. *3-1 bootstrap percolation is not a form of \mathcal{U} -bootstrap percolation.*

Proof. \mathcal{U} -bootstrap percolation is defined to be a form of bootstrap percolation where every vertex in the graph G obeys the same set of update rules, \mathcal{U} . However, note that for 3-1 BP, not all vertices have the same update rule. There are two possible \mathcal{U} here:

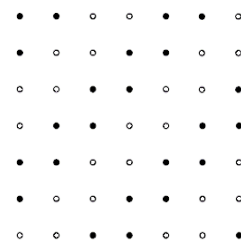


Figure 6: The lattice configuration for diagonal BP. Black vertices are label 1, white are label 2.

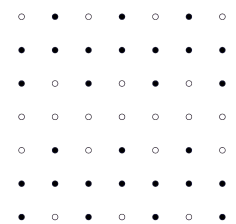


Figure 8: The lattice configuration for 3-1 BP. Black vertices are label 1, white are label 2.

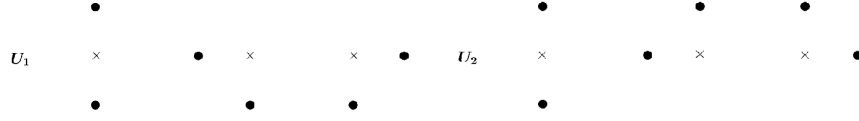


Figure 9: The update families U_1 and U_2 for 3-1 BP. There are 3 rules for each \mathcal{U} , shown here.

Thus, 3-1 bootstrap percolation is not a specific case of \mathcal{U} -bootstrap percolation. \square

Because \mathcal{T} -BP can have multiple update families \mathcal{U} , we must create a version of stable directions that applies to all \mathcal{T} -BP:

Definition 20. The direction u is a d -stable direction if there exists an $i \in [d]$ such that $[\mathbb{H}_u + i] = \mathbb{H}_u + i$, where $\mathbb{H}_u + i$ indicates a fixed half-plane \mathbb{H}_u shifted to the right i units. A direction $u \in S^1$ is a *strongly d -stable direction* if $u \in I$, where $I \subset S^1$, such that for all directions $s \in I$, s is d -stable.

Definition 21. A graph is d -critical if the set of directions D that are k -stable (where $k \leq d$), satisfy that for every open semicircle $C \subset S^1$, $C \cup D \neq \emptyset$ and there exists a semicircle $C' \subset S^1$ such that there are no strongly d -stable directions in C' . A graph is d -subcritical if every open semicircle $C \subset S^1$ contains a strongly d -stable direction.

Lemma 22. *The 0-stable directions for 3-1 bootstrap percolation are the axes. Moreover, every other direction $u \in S^1 \setminus \{i, j, -i, -j\}$ is 1-stable. Thus, 3-1 BP is 1-subcritical and 0-critical.*

Proof. The first part is clear—simply check every element of $\{i, j, -i, -j\}$ for 0-stability. Next, we focus on vectors in the first quadrant only. The proof can be extended to the other quadrants by symmetry. So, in the first quadrant, the border of any \mathbb{H}_u will have the components from Figure 5. The diagonal components imply that first, if the vertices that form the diagonal components are of the same label, then no new sites will be infected. And if the diagonal vertices are of different labels, new sites will be infected (because of the second update rule in U_1), but translate the boundary by 1 and now they have the same label—so $[\mathbb{H}_u + 1] = \mathbb{H}_u + 1$. Thus, the directions not on the axes all 1-stable. \square

3.3.2 Checkerboard BP

Begin with \mathbb{Z}^2 and assign labels to the vertices such that the graph appears as shown in Figure 10. One may note that there are two possible update families which are both subcritical: S_1 is quadrants 2 and 4, while S_2 is quadrants 1 and 3:

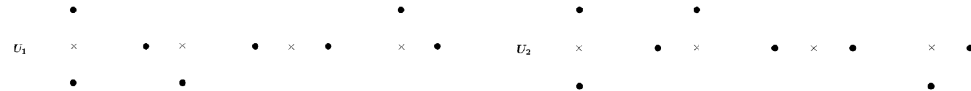
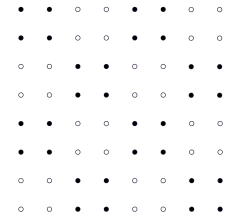


Figure 11: The update families U_1 and U_2 for checkerboard BP, with 4 update rules for each.

Figure 10: Lattice configuration for checkerboard BP. Black vertices are label 1, white are label 2.

Lemma 23. *The 0-stable directions for checkerboard bootstrap percolation are the axes. Every other direction $u \in S^1 \setminus \{i, j, -i, -j\}$ is 1-stable. Thus, checkerboard BP is 1-subcritical and 0-critical.*

Proof. This proof is nearly identical to the one for Lemma 22. The first part is clear—simply check every element of $\{i, j, -i, -j\}$ for 0-stability. Next, we focus on vectors in the first quadrant only. The proof can be extended to the other quadrants by symmetry. So, in the first quadrant, the border of any \mathbb{H}_u will have the components from Figure 5. The diagonal components imply that first, if the vertices that form the diagonal components are of the same label, then no new sites will be infected. And if the diagonal vertices are of different labels, new sites will be infected (because of the second update rule in U_1), but translate the boundary by 1 and now they have the same label—so $[\mathbb{H}_u + 1] = \mathbb{H}_u + 1$. Thus, the directions not on the axes are all 1-stable. \square

3.3.3 Mixed BP

We study this model in order to gain insight into less regular models. The network coloring is shown in Figure 13 (Right). The possible update families are:

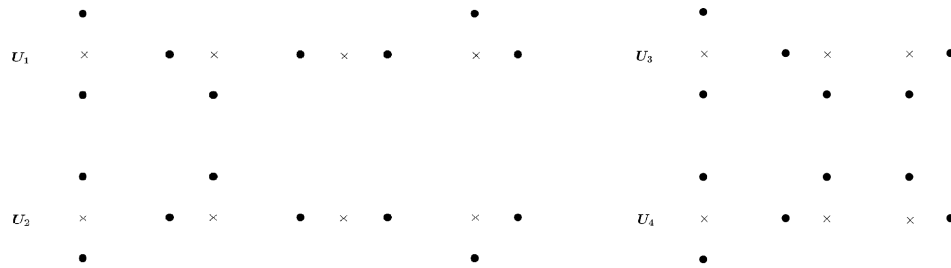


Figure 12: The update families $U_1, U_2, U_3,$ and U_4 for mixed BP. There are 4, 4, 3, and 3 rules, respectively, for each.

Note the stable directions for each \mathcal{U} : the family U_1 has stable directions in quadrants 2 and 4; the family U_2 has stable directions in quadrants 1 and 3; the family U_3 has stable directions in quadrants 3 and 4; and finally the family U_4 has stable directions in quadrants 1 and 2. In Figure 13 (Left) one can see each vertex by which update family U_i it has.



Figure 13: Left: Red vertices are U_1 , yellow are U_2 , green are U_3 , and purple are U_4 . Right: the coloring of the mixed BP, where black vertices are label 1, white are label 2.

To understand the stable directions, consider some unit vector u in the first quadrant. The vertices that can be newly infected from \mathbb{H}_u must have update families U_1 or U_3 , as quadrant 1 is unstable for these families. By symmetry, it can be seen that as long as a unit vector u has \mathbb{H}_u with a boundary such that there are 3 consecutive diagonal vertices, it is 2-stable. An obvious example would be $u = (\frac{\sqrt{2}}{2}, \frac{\sqrt{2}}{2})$. An example is displayed in Figure 14, where the process of infection is shown:

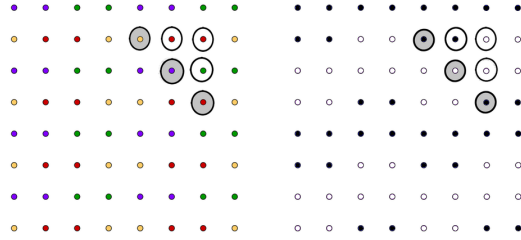


Figure 14: The left lattice contains the vertices labeled by their update family and the right has vertices labeled by their state (as in Figure 13). Suppose the vertices surrounded by gray-filled circles are initially infected. The newly infected vertices are surrounded by black circles, and note that they have update families U_1 and U_3 , as both of these families are unstable for u in the first quadrant.

3.4 Trust Model for \mathcal{T} -BP for $m = 2$ as 2-bootstrap percolation on \mathbb{Z}^2

So now we turn to the more general model: \mathcal{T} -BP for $m = 2$ as 2-bootstrap percolation on \mathbb{Z}^2 , with random assigning of labels to vertices. First, we want to be able to predict whether its growth is relatively unbounded, as in critical and supercritical models, or restricted, as in subcritical models.

Let us consider all possible update families \mathcal{U} on the model. The reason that this model does not already fall under the umbrella of \mathcal{U} -bootstrap percolation is because each of its vertices obeys a different set of update rules depending on the configuration of its neighbors, making our model non-homogeneous. Therefore, we can ask whether each possible \mathcal{U} , based on an individual vertex, is critical or subcritical. We discover that out of all possible \mathcal{U} on this model, only $\frac{1}{8}$ are critical-the rest are subcritical. From here, we arrive at the main Conjecture 16 from Section 3.1. We sketch here an approach that may allow us to ultimately prove Conjecture 16.

Definition 24. A *finitely stable graph* is one such that there exists an integer s such that every direction $u \in S^1$ is s -stable.

Conjecture 25. Every finitely stable graph has $p_c > 0$.

We hope to approach this problem with a method similar to that used in [2], as the growth of the infection would hopefully be bounded by shapes with sides perpendicular to the finitely stable directions of the model. We also define:

Definition 26. A *critical-free network* is a network in which the update family for every individual vertex is subcritical.

Conjecture 27. Every critical-free network has $p_c > 0$.

Then, if a critical-free network has $p_c > 0$, this might be used to show that \mathcal{T} -BP with $m = 2$ as 2-bootstrap percolation on \mathbb{Z}^2 in general has $p_c > 0$. This is because since the probability of a vertex being critical is only $\frac{1}{8}$, we believe the general model is sufficiently similar to the critical-free network that it would have $p_c > 0$ for similar reasoning.

4 \mathcal{T} -BP on Random Networks

Thus far, we have studied \mathcal{T} -BP on fairly regular graphs, mainly for simplicity. However, society is rarely this regular-which is why for this section, we focus on Erdős-Renyi graphs, or random graphs. The results are primarily analytical, and the codes can be found in the appendix. In particular, we study the following properties if the model:

- The initial probability of infection p ;
- The critical probability of infection p_c ;
- The “fraction percolated” (the fraction of graph infected by A_∞)
- The probability of having an edge between two vertices, denoted den , short for density.

These probabilities are considered in terms of the main variables of the \mathcal{T} -BP model: the time t ; the number m of labels a vertex may have; and the number r of different labels the set of infected neighbors must include in order for a vertex to be infected. Additionally, throughout this section, for any value requiring multiple trials, we run 30 trials. Additionally, we always look at graphs of the same size, having n (the number of vertices, $|V(G)|$) equal to 10000.

4.1 Varying Density

The first relationship we consider is the one between the fraction of a graph that would end up percolated and the initial probability of infection, p . We analyzed this for a graph with $m = 3$ and $r = 2$, with various values for den : 0.0005, 0.001, 0.002, 0.003, 0.004, 0.005. The graph is shown in Figure 15. Notably, the metastability effect (where a slight shift in p causes a large jump in the fraction percolated) is evident in this model.

Looking at an individual curve in Figure 15, one can determine the critical probability p_c for a network with $m = 3, r = 2$, and some fixed value for den . Therefore, we look at a more consolidated version of the results, as we compare the value of p_c to the value of den . We also expand our investigation to different values of m and r : we set (m, r) to be $(3, 2), (4, 2), (4, 3)$, and find the graph for p_c compared with den in each case, as shown in Figure 16.

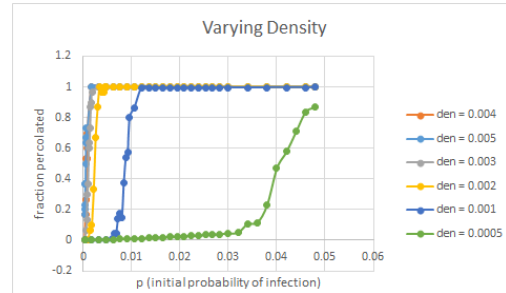


Figure 15: This shows how the fraction percolated varies as p varies given various den . Note that $n = 10000, m = 3, r = 2$. The different densities considered are shown by the differently colored curves.

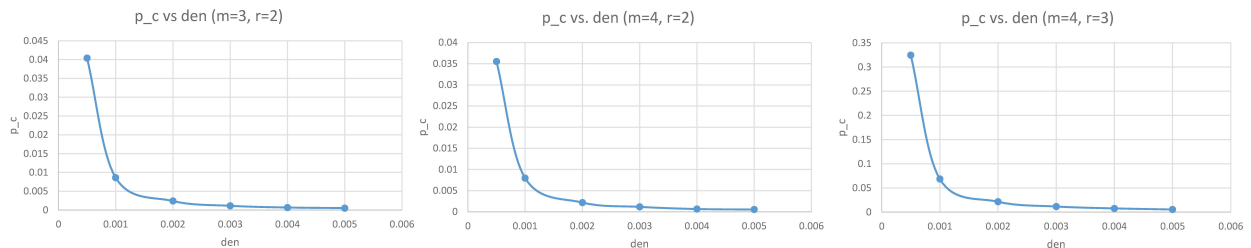


Figure 16: This figure shows how p_c varies as den varies. Note that $n = 10000$ for all cases. The pair (m, r) from left to right is $(3, 2), (4, 2), (4, 3)$.

Interestingly, this seems to follow a power law curve rather precisely. For each pair (m, r) , the curve that best fits is

- $(m, r) = (3, 2)$: $p_c = 0.000000019568213(\text{den})^{-1.897075330420950}$, with correlation coefficient $r^2 = 0.996$.
- $(m, r) = (4, 2)$: $p_c = 0.000000027712517(\text{den})^{-1.833916660925080}$, with $r^2 = 0.995$.
- $(m, r) = (3, 2)$: $p_c = 0.0000004688(\text{den})^{-1.7465029493}$, with $r^2 = 0.993$.

4.2 Varying the number of required labels

We once again look at the relationship between the fraction of a graph that would end up percolated and the initial probability of infection, p , but while varying the number of required labels, r . We begin with a graph with $m = 4$ and $\text{den} = 0.005$, with various values for r : 2, 3, 4, shown in Figure 17.

Notably, the metastability effect is clear here as well. As in the previous section, we look at each individual curve and determine the value of p_c for each r . Next, we plot r against p_c , for various pairs of parameters (m, den) set to $(4, 0.005)$, $(4, 0.003)$, $(5, 0.003)$ in Figure 18.

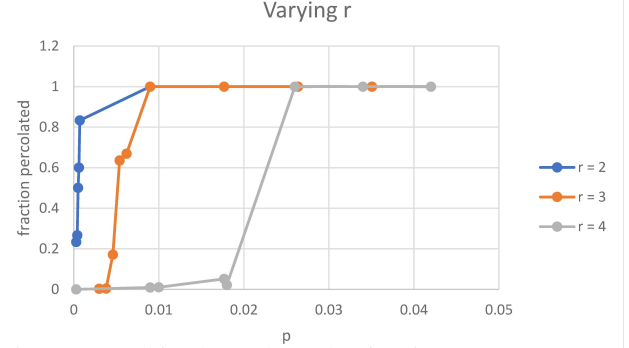


Figure 17: This shows how the fraction percolated varies as p does, for various r . Note, $n = 10000$, $m = 4$, $\text{den} = 0.005$.

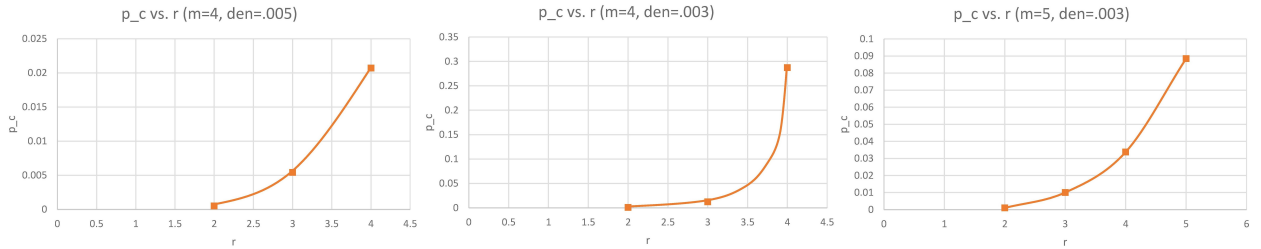


Figure 18: This figure shows how p_c varies as r varies. Note that $n = 10000$ for all cases. The pair (m, den) from left to right is $(4, 0.005)$, $(4, 0.003)$, $(5, 0.003)$.

Here, the relationship is more difficult to determine, as m and r are both small integers, and so there are only a small number of data points available to calculate. Nonetheless, it appears to fit a power law curve again, albeit more weakly in some cases. For each pair (m, den) , the curve that best fits is

- $(m, \text{den}) = (4, 0.005)$: $p_c = 0.0000135564r^{5.3410975882}$, with $r^2 = 0.997$.
- $(m, \text{den}) = (4, 0.003)$: $p_c = 0.0000030675r^{8.0395325660}$, with $r^2 = 0.975$.
- $(m, \text{den}) = (5, 0.005)$: $p_c = 0.0000390111r^{4.8654339012}$, with $r^2 = 0.994$.

Overall, p_c seems to have a power law relationship with both den and r . This suggests that perhaps the equation for p_c is of the form $p_c = cf(m)r^{e_1}\text{den}^{-e_2}$, where c, e_1, e_2 are all positive constants. Also, varying m affects p_c directly, which is why the $f(m)$ component is present.

5 Conclusion

Bootstrap percolation has been used for years to model various percolation processes, with applications spanning from epidemiology to rumor spreading. The Trust Model in Bootstrap Percolation has been developed to add an extra layer to the current forms of bootstrap percolation, as the vertices of the graphs on which the infection spreads are now labeled. The \mathcal{T} -BP was originally created to represent the spread of information based on the basic human instinct to trust information more if it comes from a variety of sources.

Using the \mathcal{T} -BP, we first looked at the probability of a vertex being immune. We discovered p_0 , or the probability of immunity, for vertices on various models: the \mathcal{T} -BP as r -bootstrap percolation, the simplest form of \mathcal{T} -BP with only one trust vector, and \mathcal{T} -BP on the deterministic hierarchical graph defined in section 2.2. Following this, we discovered the expected number of vertices which will be immune on any particular graph with these models. We also offer a lower bound for p_c , the critical probability of infection, based on p_0 . In Section 3, we looked to the \mathcal{T} -BP with $m = 2$ as 2-bootstrap percolation model on \mathbb{Z}^2 . By investigating specific “colorings” of this model (equivalently, by assigning labels to the vertices on \mathbb{Z}^2 in a non-random fashion), we were able to prove their criticality and define terminology and concepts which will hopefully be of use in proving the conjecture that $p_c > 0$ for this model. Finally, we concluded our investigation by looking to random graphs (as these better represent the irregularities of society) and deriving analytical results on these graphs by running the \mathcal{T} -BP model on them computationally.

It would be of interest to explore the \mathcal{T} -BP on different graphs. Particularly, deterministic hierarchical graphs seem promising for applications in sociology and marketing, as society has the same power law degree distribution as these graphs do. Progressing further, one may study the T-BP on random graphs with this power law degree distribution, and eventually make the model stochastic by introducing probability of the infection spreading from one vertex to another.

Additionally, the T-BP has several applications. A prevalent issue today is that of echo chambers. Social media networks and search engines keep track of news a user responds positively to, and use this information to suggest future articles and advertisements. However, this means one will increasingly only see material that is in line with one’s stated interests. This worsens issues of polarization and group-think. To combat this phenomenon, one might apply the \mathcal{T} -BP to these news-presenting algorithms. In turn, it would be interesting to acquire data from actual social networks and apply the model to it, as was done in [15]. Thus, we might ask questions such as: does the T-BP prevent the spread of fake or highly biased news? For the news that does manage to percolate through society, would it slow it down, or speed it up?

Finally, another interesting application to this model is in the study of the spread of genetic diseases. Apply the \mathcal{T} -BP to directed binary trees—representative of family trees and use 2 labels, where each label represents a possible sex. In order for a vertex to be infected, it must be infected by two vertices above it, where one vertex is a male and one is female. This represents how dominant X-linked genetic diseases are only passed on if both the male and female parent have the disease. For recessive X-linked diseases, the model would need to have 3 labels- an infected female, a female carrier, and an infected male, and would need to be stochastic (as the probability of infection from a female carrier and infected male would only be $\frac{1}{2}$).

6 Acknowledgements

This project was completed under MIT-PRIMES-USA, which the authors would like to thank for this opportunity. We would also like to thank Dr. Tanya Khovanova for her invaluable advice and Svetlana Makarova for her helpful feedback.

7 References

- [1] M. Aizenman and J. L. Lebowitz. (1988). *Metastability effects in bootstrap percolation*, J. Phys. A, *21*, 38013813.
- [2] P. Balister, B. Bollobás, M. Przykucki, and P. Smith. (2016). *Subcritical \mathcal{U} -bootstrap percolation models have non-trivial phase transitions*, Trans. Amer. Math. Soc., *368*, 7385-7411.
- [3] J. Balogh, B. Bollobás, and R. Morris. (2010). *Bootstrap percolation in high dimensions*, Combin. Probab. Comput., *19*, 643692.
- [4] J. Balogh, Y. Peres, and G. Pete. (2006). *Bootstrap percolation on infinite trees and non-amenable groups*, Combin. Probab. Comput., *15*, 715730.
- [5] J. Balogh and B. G. Pittel. (2007). *Bootstrap percolation on the random regular graph*, Random Struct. Alg., *30*, 257286.
- [6] A.L.Barabási, E. Ravaszi, T. Viscek. (2001). *Deterministic Scale-Free Networks*, Physica A: Statistical Mechanics and its Applications, *299*(3-4), 559-564.
- [7] G. Bizhani, M. Paczuski, and P. Grassberger. (2012). *Discontinuous percolation transitions in epidemic processes, surface depinning in random media, and Hamiltonian random graphs*, Phys. Rev. E, *86*, 011128.
- [8] B. Bollobás, P.J. Smith, and A.J. Uzzell. (2015). *Monotone cellular automata in a random environment*, Comin. Probab. Comput., *24*(4), 687-722.
- [9] A. Caliò and A. Tagarelli. (2018). *Trust-based dynamic linear threshold models for non-competitive and competitive influence propagation*, 17th IEEE International Conference On Trust, Security And Privacy In Computing And Communications, 156-162.
- [10] J. Chalupa, P.L. Leath, and G.R. Reich. (1979). *Bootstrap percolation on a Bethe lattice*, J. Phys. C., *12*, L31-L35.
- [11] M. Granovetter. (1978). *Threshold models of collective behavior*, American Journal of Sociology, *83*, 14201443.
- [12] J. Gravner, A. E. Holroyd, and A. Morris. (2012). *A sharper threshold for bootstrap percolation in two dimensions*, Probab. Theory Related Fields, *153*(1-2), 1-23.
- [13] A. E. Holroyd. (2003). *Sharp metastability threshold for two-dimensional bootstrap percolation*, Probab. Theory Related Fields, *125*, 195224.
- [14] S. Janson. (2009). *On percolation in random graphs with given vertex degrees*, Electron. J. Probab., *14*, 87118.
- [15] Y. Jing, L. Peiyu, T. Xiaobing, and L. Wenfeng. (2018). *Improved SIR Advertising Spreading Model and Its Effectiveness in Social Network*, Procedia Computer Science, *129*, 215-219.
- [16] P. G. Lind, L. R. da Silva, J. S. Andrade Jr., and H. J. Herrmann. (2007). *Spreading gossip in social networks*, Phys. Rev. E, *76*(3), 036117.
- [17] R. Morris,. (2011). *Zero-temperature glauber dynamics on \mathbb{Z}^d* , Probability Theory and Related Fields, *149*, 417434.
- [18] R. P. Stanley. (2011). "Inclusion-Exclusion." *Enumerative Combinatorics*, 2nd ed., Vol.1, 223-226. New York, NY: Cambridge University Press.
- [19] R. P. Stanley. (2011). "The Twelfefold Way." *Enumerative Combinatorics*, 2nd ed., Vol.1, 79-88. New York, NY: Cambridge University Press.
- [20] D. Watts. (2002). *A simple model of global cascades on random networks*, Proceedings of the National Academy of Sciences, *99*, 57665771.