Enabling and Controlling Diffusion Processes in Networks

Zhifeng Sun

Computer Science Department

Northeastern University

A thesis submitted for the degree of

PhilosophiæDoctor (PhD)

2012 April

1. Reviewer: Rajmohan Rajaraman

2. Reviewer: Ravi Sundaram

3. Reviewer: Emanuele Viola

4. Reviewer: Devavrat Shah

Day of the defense:

Signature from head of PhD committee:

Abstract

Diffusion processes are important models for many real-world phenomena, such as the spread of disease or rumors. We studied different aspects of diffusion processes in networks, focusing on designing efficient distributed algorithms for positive diffusion processes and good intervention strategies to control harmful diffusions.

First, we design and analyze various distributed algorithms for diffusion processes. We want to devise efficient distributed algorithms, which are easy to implement, to help the spreading of positive/useful information. We refer to these processes as positive diffusions. Earlier work has studied this for a variety of models, mainly based on static networks. The major point that separates our research with previous work is that we consider dynamically changing networks, which extends previous models to a larger range of real-life situations. Depending on the ways that networks are altered, we studied diffusion processes over the following two types of dynamic networks: (1) networks are changed due to individuals' decisions or behaviors; (2) networks are controlled by an adversary.

Secondly, we study how to devise good intervention strategies to control diffusion processes. This problem is crucial when we deal with harmful information like human diseases or computer viruses. We refer to these processes as harmful diffusions. We distinguish between centralized and decentralized intervention strategies. In centralized intervention strategies, there is a controller who has a limited amount of intervention resources (e.g. vaccinations or antidotes in the case of diseases). We study the problem of allocating these limited resources among the network agents so that the spread of the diffusion process is minimized. In decentralized intervention

strategies, each individual in the network makes their own decision on protecting themselves, based on their individual utility and local knowledge. In such settings, we are interested in questions such as: is there a stable set of intervention strategies? What's the cost of decentralized solutions compared with an optimal centralized one? Lastly, we augment our studies of intervention strategies with the consideration about individual behavior changes which would lead to a new kind of network dynamics. Earlier work has shown that the combination of behavior change and intervention failure (e.g. failed vaccination) can lead to perverse outcomes where less (intervention resources) is more (effective). However, the extent of the perversity and its dependence on network structure as well as the precise nature of the behavior change has remained largely unknown.

Acknowledgements

I would particularly like to thank my advisor, Rajmohan Rajaraman, for his careful proofreading and thoughtful comments as well as for serving on my thesis committee and collaborating throughout this research effort. I am also very grateful to Ravi Sundaram for his contributions to my research as well as serving on my committee, and to my co-authors on these related topics: V.S. Anil Kumar, Gopal Pandurangan, David Peleg, Chinmoy Dutta, and Bernhard Haeupler. I'd also like to thank my other committee members: Emanuele Viola, who taught me many of the techniques and tools I used in this research, and Devavrat Shah, who gave very helpful comments on both proposal and thesis. Finally, I like to thank my family and friends for their supports throughout my education.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

The process of diffusion is the spread of information or some flow in a network through local transmissions. Many real world applications can be modeled as diffusion processes over networks. Some prominent examples include diseases transmitted among humans, viruses transmitted over computer networks, information/ideas spreading over contact networks, and creation of friendships through social networks. Despite the diversity among these applications, there are fundamental similarities in the mathematical models. Understanding the properties of these applications through mathematical models can help us anticipate, exploit, and control the propagation processes.

Based on the information or nature of the commodity that is flowing, we classify diffusion processes into the following two categories, *positive diffusion* and *negative/harmful diffusion*. In positive diffusion, the information or commodities are useful to the nodes, like innovation and ideas, whereas in negative diffusion, the information or commodities are harmful to the nodes, like diseases and viruses. In positive diffusion, we are interested in analyzing the converging time and designing efficient algorithms for fast diffusion. While in negative diffusion, we are interested in analyzing the converging time and the extent of diffusion processes, as well as designing good intervention strategies. Take the spread of a disease or computer virus as an example. Lots of important questions can be asked. Will it become an epidemic? How much time does it take to become an epidemic? Who will get infected? What's the social cost of the epidemic? Once we understand all these, we can design interventions to control the dynamics. For instance, how do we vaccinate or quarantine the population so that the epidemic is controlled? How do we secure computers to enhance the network resilience?

What polices should be applied with budget constraints (limited vaccines or anti-virus software licenses), how should we distribute resources, and how much can we reduce social cost? Often these interventions can be translated into voluntary directives from government, like take vaccines or stay at home. However, people usually don't adhere to such recommendations. Instead, they make decisions based on their specific utilities and objectives. Such decisions happen in a decentralized manner, which makes game theory a natural approach to study these problems. Moreover, people alter their contacts dynamically. For example, a vaccinated person may increase his/her contacts with friends, due to perceived secure feelings. These behavioral changes have a huge impact on the dynamics and the effectiveness of these interventions, so that "good" intervention strategies might be ineffective, depending on the behavioral changes. All these make the analysis of diffusion processes more interesting and challenging.

In the first half of this dissertation, we concentrate on enabling positive diffusion. More interestingly, we focus on the diffusion processes on dynamically changing networks. The networks can be changed by the diffusion process itself or by an adversary. We introduce the problems in these two types of dynamic networks in detail in Section 1.1 and Section 1.2 respectively. In the second half of this dissertation, we switch gear to controlling negative diffusion. We design both centralized and decentralized strategies to control negative diffusion, introduced in Section 1.3. We further consider the effects of individual behavior changes on the design of control strategies, which is introduced in Section 1.4.

1.1 Diffusion under organic dynamics

Many large-scale, real-world networks such as peer-to-peer networks, the Web, and social networks are highly dynamic with continuously changing topology. The evolution of the network as a whole is typically determined by the decentralized behavior of nodes, i.e., the local topological changes made by the individual nodes (e.g., adding edges between neighbors). The dynamics can be captured as diffusion processes in self-altered networks. Understanding the dynamics of such diffusion processes is critical for both analyzing the underlying stochastic phenomena, e.g., in evolution of social networks, the Web and other real-world networks [40, 109, 123], and designing practical algorithms for associated algorithmic problems, e.g., in resource discovery in distributed networks

[75, 92] or in the analysis of algorithms for the Web [45, 51]. In this thesis, we study the dynamics of network evolution that result from *local* gossip-style processes. Gossip-based processes have recently received significant attention because of their simplicity of implementation, scalability to large network size, and robustness to frequent network topology changes; see, e.g., [54, 83, 84, 47, 82, 80, 106, 43] and the references therein. In a local gossip-based algorithm (e.g., [47]), each node exchanges information with a small number of randomly chosen neighbors in each round.[1] The randomness inherent in the gossip-based protocols naturally provides robustness, simplicity, and scalability.

We present two illustrative applications for our study. First, consider a P2P network, where nodes (computers or end-hosts with IDs/IP addresses) can communicate only with nodes whose IP address are known to them. A basic building block of such a dynamic distributed network is to efficiently discover the IP addresses of all nodes that currently exist in the network. This task, called *resource discovery* [75], is a vital mechanism in a dynamic distributed network with many applications [75, 5]: when many nodes in the system want to interact and cooperate they need a mechanism to discover the existence of one another. Resource discovery is typically done using a local mechanism [75]; in each *round* nodes discover other nodes and this changes the resulting network — new edges are added between the nodes that discovered each other. As the process proceeds, the graph becomes denser and denser and will finally result in a complete graph. Such a process was first studied in [75] which showed that a simple randomized process is enough to guarantee almost-optimal time bounds for the time taken for the entire graph to become complete (i.e., for all nodes to discover all other nodes). Their randomized *Name Dropper* algorithm operates as follows: in each round, each node chooses a random neighbor and sends *all* the IP addresses it knows. Note that while this process is also gossip based the information sent by a node to its neighbor can be extremely large (i.e., of size $\Omega(n)$).

Second, in social networks, nodes (people) discover new nodes through exchanging contacts with their neighbors (friends). Discovery of new nodes changes the underlying network — new edges are added to the network — and the process continues in the

[1] Gossip, in some contexts (see e.g., [80, 82]), has been used to denote communication with a random node in the network, as opposed to only a directly connected neighbor. The former model essentially assumes that the underlying graph is complete, whereas the latter (as assumed here) is more general and applies even to arbitrary graphs. The local gossip process is typically more difficult to analyze due to the dependencies that arise as the network evolves.

changed network. For example, consider the *LinkedIn* network[1], a large social network of professionals on the Web. The nodes of the network represent people and edges are added between people who directly know each other — between direct contacts. Edges are generally undirected, but LinkedIn also allows directed edges, where only one node is in the contact list of another node. LinkedIn allows two mechanisms to discover new contacts. The first can be thought of as a *triangulation* process (see Figure 1.1(a)): A person can introduce two of his friends that could benefit from knowing each other — he can mutually introduce them by giving their contacts. The second can be thought of as a *two-hop* process (see Figure 1.1(b)): If *you* want to acquire a new contact then you can use a shared (mutual) neighbor to introduce yourself to this contact; i.e., the new contact has to be a two-hop neighbor of yours. Both the processes can be modeled via gossip in a natural way and the resulting evolution of the network can be studied. This yields insight on the evolution of the social network over time.

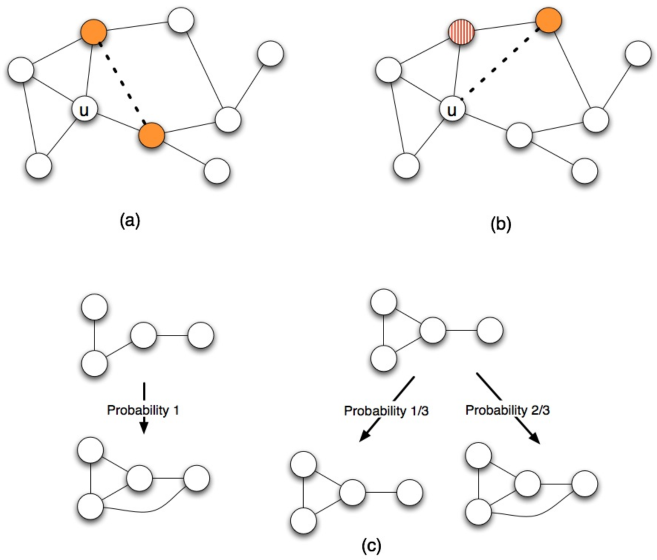


Figure 1.1: (a) Push discovery or triangulation process. (b) Pull discovery or two-hop walk process. (c) Non-monotonicity of the triangulation process – the expected convergence time for the 4-edge graph exceeds that for the 3-edge subgraph.

[1] http://www.linkedin.com.

Gossip-based discovery. Motivated by the above applications, we analyze two natural gossip-based discovery processes (also diffusion processes). We assume that we start with an arbitrary undirected connected graph and the process proceeds in synchronous rounds. Communication among nodes occurs only through edges in the network. We further assume that the size of each message sent by a node in a round is at most $O(\log n)$ bits, i.e., the size of an ID.

1. Push discovery (triangulation): In each round, each node chooses two random neighbors and connects them by "pushing" their mutual information to each other. In other words, each node adds an undirected edge between two of its random neighbors; if the two neighbors are already connected, then this does not create any new edge. Note that this process, which is illustrated in Figure 1.1(a), is completely local. To execute the process, a node only needs to know its neighbors; in particular, no two-hop information is needed.

2. Pull discovery (two-hop walk): In each round, each node connects itself to a random neighbor of one of its randomly chosen neighbors, by "pulling" a random neighboring ID from a random neighbor. Alternatively, one can think of each node doing a two-hop random walk and connecting to its destination. This process, illustrated in Figure 1.1(b), can also be executed locally: a node simply asks one of its neighbors v for an ID of one of v's neighbors and then adds an undirected edge to the received contact.

Both the above processes are local in the sense that each node only communicates with its neighbors in any round, and lightweight in the sense that the amortized work done per node is only a constant per round. Both processes are also easy to implement and generally oblivious to the current topology structure, changes or failures. It is interesting also to consider variants of the above processes in directed graphs. In particular, we study the two-hop walk process which naturally generalizes in directed graphs: each node does a two-hop directed random walk and adds a *directed* edge to its destination. We are mainly interested in the time taken by the process to converge to the transitive closure of the initial graph, i.e., till no more new edges can be added.

Our results. We present almost-tight bounds on the number of rounds it takes for the push and pull discovery processes to converge.

- **Undirected graphs:** In Sections 2.2 and 2.3, we show that for *any* undirected n-node graph, both the push and the pull discovery processes converge in $O(n \log^2 n)$ rounds with high probability. We also show that $\Omega(n \log n)$ is a lower bound on the number of rounds needed for almost any n-node graph. Hence our analysis is tight to within a logarithmic factor.

- **Directed graphs:** In Section 2.4, we show that the pull process takes $O(n^2 \log n)$ time for any n-node directed graph, with high probability. We show a matching lower bound for weakly connected graphs, and an $\Omega(n^2)$ lower bound for strongly connected directed graphs. Our analysis indicates that the directionality of edges can greatly impede the resource discovery process.

Applications. The gossip-based discovery processes we study are directly motivated by the two scenarios outlined above, namely algorithms for resource discovery in distributed networks and analyzing how discovery process affects the evolution of social networks. Since our processes are simple, lightweight, and easy to implement, they can be used for resource discovery in distributed networks. The original resource discovery algorithm of [75] was helpful in developing systems like Akamai. Unlike prior algorithms for the discovery problem [75, 92, 91, 5], the amortized work done per node in our processes is only constant per round and hence this can be efficiently implemented in bandwidth and resource-constrained networks (e.g., peer-to-peer or sensor networks). In contrast, the *Name Dropper* algorithm of [75], can transfer up to $\Theta(n)$ information per edge per round and hence may not be scalable for large-scale networks. We note that, however, because there is essentially no restriction on the bandwidth, the number of rounds taken by the *Name Dropper* algorithm is $O(\log^2 n)$. (We note that in our model, $\Omega(n)$ is a trivial lower bound). Our analyses can also give insight into the growth of real-social networks such as LinkedIn, Twitter, or Facebook, that grow in a decentralized way by the local actions of the individual nodes. For example, it can help in predicting the sizes of the immediate neighbors as well as the sizes of the second and third-degree neighbors (e.g., these are listed for every node in LinkedIn).

An estimate of these can help in designing efficient algorithms and data structures to search and navigate the social network.

Technical contributions. Our main technical contribution is a probabilistic analysis of localized gossip-based discovery in arbitrary networks. While our processes can be viewed as graph-based coupon collection processes, one significant distinction with past work in this area [6, 11, 56] is that the graphs in our processes are constantly changing. The dynamics and locality inherent in our process introduces nontrivial dependencies, which makes it difficult to characterize the network as it evolves. A further challenge is posed by the fact that the expected convergence time for the two processes is *not monotonic*; that is, the processes may *take longer* to converge starting from a graph G than starting from a subgraph H of G. Figure 1.1(c) presents a small example illustrating this phenomenon. This seemingly counter-intuitive phenomenon is, however, not surprising considering the fact that the cover time of random walks also share a similar property. One consequence of these hurdles is that analyzing the convergence time for even highly specialized or regular graphs is challenging since the probability distributions of the intermediate graphs are hard to specify. Our lower bound analysis for a specific strongly connected directed graph in Theorem 15 illustrates some of the challenges. In our main upper bound results, we overcome these technical difficulties by presenting a uniform analysis for all graphs, in which we study different local neighborhood structures and show how each lead to rapid growth in the minimum degree of the graph.

1.2 Diffusion under adversarial dynamics

In an adversarial dynamic network, nodes and communication links can appear and disappear at will over time. Emerging networking technologies such as ad hoc, wireless, sensor, mobile, and peer-to-peer networks are inherently dynamic, resource-constrained, and unreliable. This necessitates the development of a solid foundation to design efficient, robust, and scalable algorithms for diffusion processes in adversarial networks, and to understand the power and limitation of distributed computing on such networks. Such a foundation is critical to realize the full potential of these large-scale dynamic communication networks.

As a step towards understanding the fundamental computation power of such dynamic networks, we investigate dynamic networks in which the network topology changes arbitrarily from round to round. We first consider a worst-case model that was introduced by Kuhn, Lynch, and Oshman [89] in which the communication links for each round are chosen by an online adversary, and nodes do not know who their neighbors for the current round are before they broadcast their messages. (Note that in this model, only edges change and nodes are assumed to be fixed.) The only constraint on the adversary is that the networks should be connected at each round. Unlike prior models on dynamic networks, the model of [89] does not assume that the network eventually stops changing and requires that the algorithms work correctly and terminate even in networks that change continually over time.

We study a fundamental diffusion process, information spreading (also known as gossip) in such dynamic network. In gossip, or more generally, k-gossip, there are k pieces of information (or tokens) that are initially present in some nodes and the problem is to disseminate the k tokens to all nodes. By just gossip, we mean n-gossip, where n is the network size. Information spreading is a fundamental primitive in networks which can be used to solve other problems such as leader election.

The focus of this thesis is on the power of *token-forwarding* algorithms, which do not manipulate tokens in any way other than storing and forwarding them. Token-forwarding algorithms are simple, often easy to implement, and typically incur low overhead. In a key result, [89] showed that under their adversarial model, k-gossip can be solved by token-forwarding in $O(nk)$ rounds, but that any deterministic online token-forwarding algorithm needs $\Omega(n \log k)$ rounds. They also proved an $\Omega(nk)$ lower bound for a special class of token-forwarding algorithms, called knowledge-based algorithms. Our main result is a new lower bound on *any* deterministic online token-forwarding algorithm for k-gossip.

- We show that every deterministic online token-forwarding algorithm for the k-gossip problem takes $\Omega(nk/\log n)$ rounds. Our result applies even to centralized (deterministic) token-forwarding algorithms that have a global knowledge of the token distribution.

This result resolves an open problem raised in [89], significantly improving their lower bound, and matching their upper bound to within a logarithmic factor. Our lower bound also enables a better comparison of token-forwarding with an alternative approach based on network coding due to [72, 73], which achieves a $O(nk/\log n)$ rounds using $O(\log n)$-bit messages (which is not significantly better than the $O(nk)$ bound using token-forwarding), and $O(n+k)$ rounds with large message sizes (e.g., $\Theta(n \log n)$ bits). It thus follows that for large token and message sizes there is a factor $\Omega(\min\{n, k\}/\log n)$ gap between token-forwarding and network coding. We note that in our model we allow only one token per edge per round and thus our bounds hold regardless of the token size.

Our lower bound indicates that one cannot obtain efficient (i.e., subquadratic) token-forwarding algorithms for gossip in the adversarial model of [89]. Furthermore, for arbitrary token sizes, we do not know of any algorithm that is significantly faster than quadratic time. This motivates considering other weaker (and perhaps, more realistic) models of dynamic networks. In fact, it is not clear whether one can solve the problem significantly faster even in an offline setting, in which the network can change arbitrarily each round, but the entire evolution is known to the algorithm in advance. Our next contribution takes a step in resolving this basic question for token-forwarding algorithms.

- We present a polynomial-time offline token-forwarding algorithm that solves the k-gossip problem on an n-node dynamic network in $O(\min\{nk, n^{1.5}\sqrt{\log n}\})$ rounds.

- We also present a polynomial-time offline token-forwarding algorithm that solves the k-gossip problem in a number of rounds within an $O(n^\epsilon)$ factor of the optimal, for any $\epsilon > 0$, assuming the algorithm is allowed to transmit $O(\log n)$ tokens per round.

The above upper bounds show that in the offline setting, token-forwarding algorithms can achieve a time bound that is within $O(\sqrt{n \log n})$ of the information-theoretic lower bound of $\Omega(n+k)$, and that we can approximate the best token-forwarding algorithm to within a $O(n^\epsilon)$ factor, given logarithmic extra bandwidth per edge.

1.3 Controlling negative diffusion

In this section, we motivate our problems using computer virus as an example. However the study of intervention strategies can be easily applied to other fields like epidemiology.

Over the recent decades, there has been an explosive growth in the use of personal digital devices of various kinds, which are connected to the Internet through new technologies, such as Bluetooth and Wi-Fi to allow ubiquitous access. This has, unfortunately, been accompanied by significant increase in worm attacks that exploit bugs in these new technologies, and which have new and growing "medium" to spread on - recent attacks, e.g., Cabir and CommWorm, that span multiple networks are expected to become increasingly prevalent in future. While, effective anti-virus software and patches are readily available, the average user is very independent and does not often care to be proactive about installing the most effective anti-virus software, and downloading the latest patches, partially because of the cost of the software and the effort involved, which we refer to as the *security cost.* Indeed, a large fraction of devices are estimated to be without adequate anti-virus protection. If a user does not install protective software, they would incur a cost if his device gets attacked, due to downtime, loss of revenue, and cost of re-installing systems; we refer to these as the *infection cost.* If enough other nodes in the network are secured, the likelihood of a specific device getting infected would go down (as a result of the "herd immunity"), leading to a natural game theoretic scenario. A number of different non-cooperative game formulations have been developed to study this basic problem, e.g., [15, 16, 35, 67, 71, 94, 126]; one issue with many of these formulations is that they involve utility functions that require quite a lot of non-local information to compute, and it is not clear how implementable such games might be.

In this thesis, we present a generalized network security game model GNS(d), which incorporates arbitrary contact networks through an undirected graph G and heterogeneous nodes with individual security and infection costs. Our model is parametrized by network locality parameter d, which represents the distance within the network that a given infection can spread. Equivalently, the parameter d in the game GNS(d) could represent the extent of neighborhood information that is available to a node when making strategic security decisions, which is a departure from earlier models which require global information for making decisions. Qualitatively, we consider three important cases with respect to d. The case $d = 1$, which we refer to as the *local infection model*, is most well-suited for ad hoc wireless networks and social networks, when certain actions initiated by an insecure node could adversely affect immediate neighbors, friends, or email contacts. For this case, our model can be viewed as a variant of the IDS model of [81]. The case $d = \infty$, which we refer to as the *global infection model*, is most well-suited for the highly infectious worms and viruses in the Internet that can be transmitted in an hop-unlimited manner through unsuspecting insecure nodes, under the assumption that individual nodes have complete information. Our GNS(∞) model is a generalization of the elegant model of [15]. The intermediate case $1 < d < \infty$ applies to the majority of network security hazards where the transmission may be hop-limited and nodes may only have limited local information about the topology and security decisions taken by others. Our main results are the following.

- **Existence of pure Nash equilibria (NE):** We show that the locality parameter d plays a significant role in the structure of the resulting games. Both the extremes of GNS(1) and GNS(∞) turn out to be ordinal potential games, and a pure NE can be computed by best response dynamics – that is, every sequence of best response steps by the individual players converges to a pure NE. However, for every d in the range $(1, \infty)$, there exists an instance of GNS(d) that does not have a pure equilibrium. The price of anarchy for a GNS(1) game is at most the maximum degree of the contact graph, while that for GNS(∞) is inversely proportional to the vertex expansion of the contact graph.

- **Complexity of computing pure NE:** While there is a simple combinatorial characterization for the existence of pure NE in GNS(d) for all d, we show that for $1 < d < \infty$, deciding if an arbitrary instance of GNS(d) has a pure NE is NP-complete. For GNS(1), we show that finding a pure NE of least cost is NP-complete; a corresponding result for GNS(∞) is in [15].

- **Approximating the social optimum:** We show that computing the social optimum is NP-complete for a GNS(d) game, for any d; the case of $d = \infty$ was shown by [15]. We design a general framework for finding a strategy vector for the players in polynomial time, whose cost is at most $2d$ times that of the optimal, for any fixed d. In particular, this implies that for $d = 1$, we obtain a 2-approximation. For $d = \infty$, we provide a different algorithm within the framework that yields an $O(\log n)$-approximation, where n is the number of nodes in the network; this improves on the approximation bound of $O(\log^{1.5} n)$ of [15] achieved for a special case of the GNS(∞).

- **Empirical results:** We study the characteristics of NE empirically in two distinct classes of graphs: random geometric graphs and power law graphs. For $d = 1$, we find that the convergence time for best response is sub-linear in the number of nodes in both the classes of graphs, while it is linear for $d = \infty$. Also, for $d = 1$, we find that the cost of the pure NE obtained is very close to that of the social optimum, indicating that the pure NE obtained in real-world networks approximate social optimum very well. For $d = \infty$, we observe that there may be a significant gap between the cost of the pure NE and that of the social optimum, even for small networks. Finally, we study the performance of our approximation algorithms for the social optimum, and find that the approximation guarantees in practice are much smaller than our theoretical bounds.

Pure NE represent stable operating points for a system with selfish users. Therefore, for a network planner, understanding and controlling the quality of equilibria reached is an important issue. Our results suggest locality characteristics of the network or the amount of information available to the strategic network players have a significant impact on the existence of equilibria. The non-monotonicity in the existence of NE, with respect to d, is somewhat surprising and suggests a closer examination of the impact of information on pure NE in such games. While our theoretical analysis indicates that pure NE may be significantly inferior to the optimum in terms of social optimum in the worst-case, our experiments suggest that for real-world network models pure NE obtained by uncoordinated best response dynamics have low cost relative to the social optimum, especially in the case of $d = 1$. Additionally, our results on the price of anarchy suggest natural heuristics to aid a network planner in enforcing efficient equilibria. Finally, the approximations achieved by our approximation algorithms, both in theory and experiments, indicate that our proposed algorithms are viable candidates wherever centralized decisions can be made on network protection mechanisms.

1.4 Controlling negative diffusion in the presence risk behavior changes

The study in Section 1.3 assumes that the behavior of each individual remains the same before and after taking interventions. However, this is not an accurate assumption in some real world scenarios. For instance, people expose themselves more to the public after taking vaccinations. This behavior change is often referred as risk behavior change. And it is very common, specially in epidemiology. Since vaccination is not 100% reliable, this kind of behavior change has the potential to increase the likelihood of disease transmission. In our study, it is important to consider the impact of risk behavior to good intervention strategies. In our discussion below, we use disease transmission as example, but risk behavior is not limited to epidemiology.

For many diseases, such as influenza and HIV, prophylactic interventions using anti-virals and vaccinations are commonly used to control the spread of the diseases, and are usually universally recommended, barring individual constraints. Recent studies have shown significant benefits of anti-retrovirals for reducing the spread of HIV [61]. Such treatments have varying levels of efficacy (25-75% in the case of HIV [61, 70] and between 10-80% in the case of influenza [1], depending on the demographics and the specifics of the flu strain). However, people are not very well aware of this limitation, and studies often over-estimate the efficacy of vaccines [114]. Indeed, the perceived protection from infection might cause behavioral changes, leading to an increase in contact by a treated individual; such a behavioral change following vaccination could also be a natural evolutionary response [86, 105], and has also been documented recently in the context of flu vaccines [115]. Regardless of the underlying reasons, failure of prophylactic interventions in conjunction with increased social behavior can have significant unexpected effects on the disease dynamics. In a series of important papers [37, 98] Blower and her collaborators demonstrated that risk behavior change, in the context of HIV vaccinations, could lead to perverse outcomes. Subsequently, several independent studies have confirmed this phenomenon of perversity in the use of HIV vaccines and anti-virals, and vaccines for the human papillomavirus (HPV) [119, 36, 128, 124, 70, 121, 13, 52, 74, 100, 44, 64].[1]

A fundamental question in mathematical epidemiology is to determine the fraction of the population that needs to be vaccinated or treated with anti-virals in order to minimize the impact of the disease, especially when the supply is limited. Modern epidemiological analysis is largely based on an elegant class of models, called SIR

[1] The phenomenon of an increase in risky behavior following protection is also referred to as "moral hazard" and has been studied extensively in a number of areas, such as insurance (e.g., [103]); in the epidemiology literature, this is referred to more commonly as "risk behavior" (e.g., [37]), and we will fix on this terminology for most of the thesis.

(susceptible-infected-recovered), which was first formulated by Reed and Frost in the 1920s, and developed over the subsequent decades. The SIR model and its variants have been highly influential in the study of epidemics [129, 97, 99, 79, 69, 98]. These models, however, do not attempt to capture the rich structure of the contact network over which interactions occur. Network structure has a direct effect on both the spread of diseases as well as the nature of interactions, which has been observed by a number of researchers, e.g. [108, 76]. In the emerging area of contact network epidemiology, an underlying contact graph captures the patterns of interactions which lead to the transmission of a disease [113, 55, 95, 101, 102, 110, 127, 67]. Many studies have predicted the spread of diseases through networks using mathematical analysis or simulations. As we have argued above, moral-hazarding/risky behavior clearly plays an important role in the effectiveness of such interventions. While the impact of risky behavior on prophylactic treatments has been studied in previous work, the extent of the perversity and its dependence on network structure as well as the precise nature of the behavior change has remained largely unknown. [1]

In this thesis, we study the impact of risk behavior change on the spread of diseases in networks and observe a rich and complex structure dependent both on the underlying network characteristics as well as the nature of the change in behavior. We use a discrete-time SIR model of disease transmission on a contact network. The contact network is an undirected graph with each edge having a certain probability of disease transmission. An infected node is assumed to recover in one time step. We consider both uniform random vaccination (where each node is vaccinated independently with the same probability) as well as targeted vaccinations (where nodes are vaccinated based on their degree of connectedness). Vaccines are assumed to fail uniformly and randomly.[2] We model risk behavior change by an increase in the disease transmission probability. A significant aspect of our work is the consideration of the "sidedness" of risk behavior change. We classify risk behavior as one-sided or two-sided based on whether the increase in disease transmission probability requires an increase in

[1] Similar issues arise in the context of the spread of malware through infected computers. Several studies, e.g., [2], have found that computer and smart-phone users do not relate bot infections to risky behavior, such as downloading spam mails, though a large fraction of users have updated anti-virus software. It is plausible that such phenomena can also be associated with risky behavior in many cases.

[2] Though we focus on vaccinations and disease transmission, the basic results apply to other prophylactic treatments such as anti-virals, and other phenomena such as malware spread.

risk behavior of both the infector and the infectee or just the infector. As examples: influenza (H1N1) may be modeled as a one-sided disease since a vaccinated individual may be motivated to behave more riskily (going to crowded places, traveling on planes etc.,) thus increasing the chance of infecting all the individual comes in contact with; whereas AIDS (HIV) may be modeled as a two-sided disease since the increase in disease transmission requires both the individuals participating in the interaction to engage in risky behavior. Of course, these examples are simplistic and most diseases have elements of both one-sided as well as two-sided risk behavior.

Our main findings are threefold.

- First, we find that the *severity of the epidemic varies non-monotonically as a function of the vaccinated fraction.* The specific dynamics depend on the nature of risky behavior, as well as the efficacy of the vaccine (the less reliable the vaccine, the greater the non-monotonicity) and the contagiousness of the disease, but in general, we observe that increased vaccination does not immediately imply less severity; in some cases, the severity could increase by as much as a factor of two.

- Second, we find that *one-sided risk behavior change leads to perverse outcomes at low levels of vaccination, while two-sided risk behavior change leads to perverse outcomes at high levels of vaccination.* Our analysis indicates that effective prophylactic interventions against diseases with one-sided risk behavior change need to have sufficiently high coverage; on the other hand, for diseases with two-sided risk behavior change, it is essential to combine prophylactic treatments with education programs aimed at reducing risky behavior.

- Our third and, perhaps, most surprising finding is that *interventions that target highly connected individuals can be strictly worse than random interventions* for the same level of coverage and that this phenomenon occurs both for one-sided as well as two-sided risk behavior change. Given prior work on targeting vaccine distributions, this finding flies in the face of intuition that expects that targeted vaccination would confer greater benefits.

Our results have direct implications for public policy on containing epidemic spread through prohylactic interventions. Implications of risk behavior in public health have been examined earlier, e.g. [37, 98]. These prior studies are based on differential equation models, which divide the population into a fixed set of groups and model the interaction between different groups in a uniform way. The epidemic spread is then characterized by the "reproductive number," denoted by R_0, with the expected epidemic size exhibiting a threshold behavior in terms of R_0. In contrast, we use a network model that captures the fine structure of interactions between (an arbitrary number of) individuals rather than (a fixed set of) groups, and find that the network structure has a significant impact on the resulting dynamics. The heterogenous network model extends to a larger range of real-life situations but the increased fidelity comes at a price. The outcomes are more complicated and varied and the general approach of lowering R_0 does not appear to be directly applicable. Another new contribution of our study is the focus on the sidedness inherent in risk behavior change, which has not been considered before. Prior research has implicitly assumed one-sided risk behavior change where vaccinated individuals engage in risky behavior increasing the chances of infection of those they come in contact with. This work explicitly treats both one-sided and two-sided risk behavior changes and shows that their differing impact needs to be considered in public intervention policies.

1.5 Overview

In this thesis, we design efficient algorithms to enable positive diffusion and good intervention strategies to control negative diffusion. In Chapter 2, we study two nature diffusion processes under organic dynamics, and show an almost tight upper bound for both of these processes. In Chapter 3, we study similar problems as in Chapter 2, but under adversarial dynamics. We show an lower bound for any token-forwarding algorithms under online adversarial model, and design two efficient algorithms under offline adversarial model. In Chapter 4, we study both centralized and decentralized intervention strategies. We give an $O(\log n)$ approximation algorithm for optimal centralized intervention strategy. Then we show the existence of intervention strategies in decentralized settings and compare their costs with the optimal centralized strategy. In Chapter 5, we extend the study in Chapter 4 to the presence of risk behavior changes, and observe two interesting phenomena: 1) less interventions can be more effective, and 2) targeted intervention strategy can be worse than random intervention strategy. We conclude in Chapter 6.

Chapter 2

Diffusion under organic dynamics

In this chapter, we study diffusion processes under organic dynamics that are motivated by information discovery in large-scale distributed networks such as peer-to-peer and social networks. A well-studied problem in peer-to-peer networks is *resource discovery*, where the goal for nodes is to discover all other nodes in the network. For example, a node may want to know the IP addresses of all the other hosts in the network. In social networks, nodes (people) discover new nodes through exchanging contacts with their neighbors (friends). In both cases the discovery of new nodes changes the underlying network — new edges are added to the network — and the process continues in the changed network.

We study and analyze two natural gossip-based diffusion/discovery processes. In the *push discovery* or *triangulation* process, each node repeatedly chooses two random neighbors and connects them (i.e., "pushes" their mutual information to each other). In the *pull discovery* process or the *two-hop walk*, each node repeatedly requests or "pulls" a random contact from a random neighbor and connects itself to this two-hop neighbor. Both processes are lightweight in the sense that the amortized work done per node is constant per round, local, and naturally robust due to the inherent randomized nature of gossip.

Our main result is an almost-tight analysis of the time taken for these two randomized processes to converge. We show that in any undirected n-node graph both processes take $O(n \log^2 n)$ rounds to connect every node to all other nodes with high probability, whereas $\Omega(n \log n)$ is a lower bound. We also study the two-hop walk in directed graphs, and show that it takes $O(n^2 \log n)$ time with high probability, and that the worst-case bound is tight for arbitrary directed graphs, whereas $\Omega(n^2)$ is a lower bound for strongly connected directed graphs. A key technical challenge that we overcome in our work is the analysis of a randomized process that itself results in a constantly changing network leading to complicated dependencies in every round.

In Section 2.1 we list the notations and prove some common lemmas we will use in the proofs. We show the upper and lower bounds of the push discovery and the pull discovery in Section 2.2 and 2.3 respectively. Then we give the proofs of upper and lower bound of the pull discovery in directed graph in Section 2.4. Finally, we conclude in Section 2.5.

2.1 Preliminaries

In this section, we define the notations used in our proofs, and prove some common lemmas for Section 2.2 and Section 2.3. Let G denote a connected graph, $d(u)$ denote the degree of node u, and $N^i(u)$ denote the set of nodes that are at distance i from u. Let δ denote the minimum degree of G. We note that G, $d(u)$, and $N^i(u)$ all change with time, and are, in fact, random variables. For any nonnegative integer t, we use subscript t to denote the random variable at the start of time t; for example G_t refers to the graph at the start of step t. For convenience, we list the notations in Table 2.1.

Table 2.1: Notation table

Notation	description
δ_t	minimum degree of graph G_t
$N_t^i(u)$	set of nodes that are at distance i from u in G_t
$\lvert N_t^i(u)\rvert$	number of nodes in $N_t^i(u)$
$d_t(u)$	degree of node u in G_t
$d_t\left(u, N_t^i(v)\right)$	number of edges from u to nodes in $N_t^i(v)$, i.e., degree induced on $N_t^i(v)$

We present two lemmas that are used in the proofs in Section 2.2 and Section 2.3. Lemma 1 gives a lower bound on the number of neighbors within distance 4 for any node u in G_t while Lemma 2 is a standard analysis of a sequence of Bernoulli experiments.

Lemma 1. $\left|\cup_{i=1}^4 N_t^i(u)\right| \geq \min\{2\delta_t, n-1\}$ *for all* u *in* G_t.

Proof. If $N_t^3(u)$ is not an empty set, consider node $v \in N_t^3(u)$. Since $d_t(v) \geq \delta_t$, we have $\left|\cup_{i=2}^4 N_t^i(u)\right| \geq \delta_t$. $\left|N_t^1(u)\right| \geq \delta_t$ because $d_t(u) \geq \delta_t$. We also know $N_t^1(u)$ and $\cup_{i=2}^4 N_t^i(u)$ are disjoint. Thus, $\left|\cup_{i=1}^4 N_t^i(u)\right| \geq 2\delta_t$. If $N_t^3(u)$ is an empty set, then $N_t^1(u) \cup N_t^2(u) = n-1$ because G_t is connected. Thus $\left|\cup_{i=1}^4 N_t^i(u)\right| = n-1$. Combine the above 2 cases, we complete the proof of this lemma. □

Lemma 2. *Consider k Bernoulli experiments, in which the success probability of the ith experiment is at least i/m where $m \geq k$. If X_i denotes the number of trials needed for experiment i to output a success and $X = \sum_{i=1}^k X_i$, then*

$$\Pr[X > (c+1)n \ln n] < \frac{1}{n^c}$$

Proof. Since X only increases with k, with out loss of generality assume that $k = m$. Now we can view this as *coupon collector problem* [104] where X_{m+1-i} is the number of steps to collect the ith coupon. Consider the probability of not obtaining the ith coupon after $(c+1)n \ln n$ steps. This probability is

$$\left(1 - \frac{1}{n}\right)^{(c+1)n \ln n} < e^{-(c+1)\ln n} = \frac{1}{n^{c+1}}$$

By union bound, the probability that some coupon has not been collected after $(c+1)n \ln n$ steps is less than $1/n^c$. And this completes the proof of this lemma. □

2.2 The triangulation: Discovery through push

In this section, we analyze the triangulation process on undirected connected graphs, which is described by the following simple iteration: In each round, for each node u, we add edge (v, w) where v and w are drawn uniformly at random from $N_t^1(u)$. The triangulation process yields the following push-based resource discovery protocol. In each round, each node u introduces two random neighbors v and w to one another. The main result of this section is that the triangulation process transforms an arbitrary connected n-node graph to a complete graph in $O(n \log^2 n)$ rounds with high probability. We also establish an $\Omega(n \log n)$ lower bound on the triangulation process for almost all n-node graphs.

2.2.1 Upper bound

We obtain the $O(n \log^2 n)$ upper bound by proving that the minimum degree of the graph increases by a constant factor (or equals $n-1$) in $O(n \log n)$ steps. Towards this objective, we study how the neighbors of a given node connect to the two-hop neighbors of the node. We say that a node v is ***weakly tied*** to a set of nodes S if v has less than $\delta_0/2$ edges to S (i.e., $d_t(v,S) < \delta_0/2$), and ***strongly tied*** to S if v has at least $\delta_0/2$ edges to S (i.e., $d_t(v,S) \geq \delta_0/2$). Recall that δ_0 is the minimum degree at start of round 0. Then, we have the following two lemmas.

Lemma 3. *If $\delta_0 \leq d_t(u) < (1+1/4)\delta_0$ and $w \in N_0^1(u)$ is strongly tied to $N_t^2(u)$, then the probability that u connects to a node in $N_t^2(u)$ through w in round t is at least $2/(7n)$.*

Proof. Since w is strongly tied to $N_t^2(u)$, $d_t\left(w, N_t^2(u)\right) \geq \delta_0/2$. Therefore, the probability that u connects to a node in $N_t^2(u)$ through w in round t is

$$\begin{aligned}
&= \frac{d_t\left(w, N_t^2(u)\right)}{d_t(w)} \cdot \frac{1}{d_t(w)} \geq \frac{d_t\left(w, N_t^2(u)\right)}{d_t(w)} \cdot \frac{1}{n} \geq \frac{d_t\left(w, N_t^2(u)\right)}{|N_t^1(u)| + d_t\left(w, N_t^2(u)\right)} \cdot \frac{1}{n} \\
&\geq \frac{d_t\left(w, N_t^2(u)\right)}{(1+1/4)\delta_0 + d_t\left(w, N_t^2(u)\right)} \cdot \frac{1}{n} \geq \frac{\delta_0/2}{(1+1/4)\delta_0 + \delta_0/2} \cdot \frac{1}{n} = \frac{2}{7n}.
\end{aligned}$$

□

Lemma 4. *If $\delta_0 \leq d_t(u) < (1+1/4)\delta_0$, $w \in N_0^1(u)$ is weakly tied to $N_t^2(u)$, and $v \in N_0^2(u) \cap N_0^1(w)$, then the probability that u connects to v through w in round t is at least $1/(4\delta_0^2)$.*

Proof. Since w is weakly tied to $N_t^2(u)$, we know that $d_t(w)$ equals $|N_t^1(u)| + d_t\left(w, N_t^2(u)\right)$, which is at most $(1+1/4)\delta_0 + \delta_0/2$. Therefore, the probability that u connects to v through w in round t is

$$= \frac{1}{d_t(w)^2} \geq \frac{1}{((1+1/4)\delta_0 + \delta_0/2)^2} \geq \frac{1}{(7\delta_0/4)^2} \geq \frac{1}{4\delta_0^2}.$$

□

For analyzing the growth in the degree of a node u, we consider two overlapping cases. The first case is when more than $\delta_0/4$ nodes of $N_t^1(u)$ are strongly tied to $N_t^2(u)$, and the second is when less than $\delta_0/3$ nodes of $N_t^1(u)$ are strongly tied to $N_t^2(u)$. The analysis for the first case is relatively straightforward: when several neighbors of a node u are strongly tied to u's two-hop neighbors, then their triangulation steps connect u to a large fraction of these two-hop neighbors.

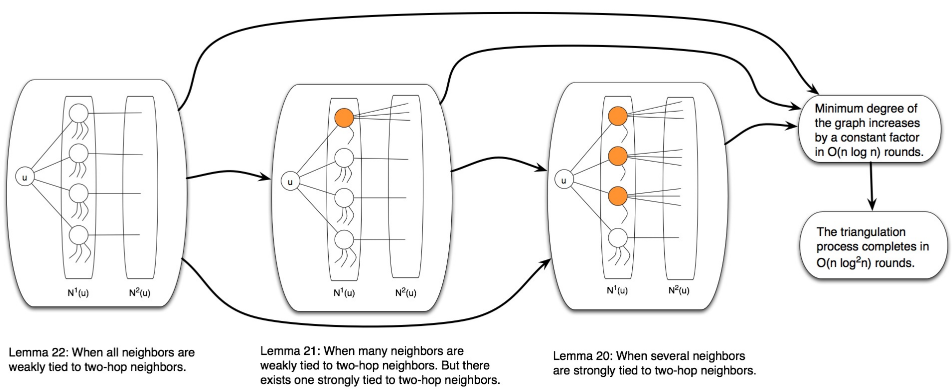

u
$N^1(u)$
$N^2(u)$
u
$N^1(u)$
$N^2(u)$
u
$N^1(u)$
$N^2(u)$
Minimum degree of the graph increases by a constant factor in O(n log n) rounds.
The triangulation process completes in O(n log²n) rounds.
Lemma 22: When all neighbors are weakly tied to two-hop neighbors.
Lemma 21: When many neighbors are weakly tied to two-hop neighbors. But there exists one strongly tied to two-hop neighbors.
Lemma 20: When several neighbors are strongly tied to two-hop neighbors.

Figure 2.1: This figure illustrates the different cases and relations between lemmas used in the proof of Theorem 8. The shaded nodes in $N_t^1(u)$ are strongly tied to $N_t^2(u)$. Others are weakly tied to $N_t^2(u)$.

Lemma 5 (When several neighbors are strongly tied to two-hop neighbors). *There exists $T = O(n \log n)$ such that if more than $\delta_0/4$ nodes in $N_t^1(u)$ are strongly tied to $N_t^2(u)$ for all $t < T$, then $d_T(u) \geq (1+1/4)\delta_0$ with probability at least $1-1/n^2$.*

Proof. If at any round $t < T$, $d_t(u) \geq (1 + 1/4)\,\delta_0$, then the claim of the lemma holds. In the remainder of this proof, we assume $d_t(u) < (1 + 1/4)\,\delta_0$ for all $t < T$. Let $w \in N_t^1(u)$ be a node that is strongly tied to $N_t^2(u)$. By Lemma 3 we know that

$$\Pr\left[u \text{ connects to a node in } N_t^2(u) \text{ through } w \text{ in round } t\right] \geq \frac{2}{7n} > \frac{1}{6n}$$

We have more than $\delta_0/4$ such w's in $N_t^1(u)$, each of which independently executes a triangulation step in any given round. Consider a run of $T_1 = 72n \ln n/\delta_0$ rounds. This implies at least $18n \ln n$ attempts to add an edge between u and a node in $N_t^2(u)$. Thus,

$$\begin{aligned}
&\Pr\left[u \text{ connects to a node in } N_t^2(u) \text{ after } T_1 \text{ rounds}\right] \\
\geq\ & 1 - \left(1 - \frac{1}{6n}\right)^{18n \ln n} \\
\geq\ & 1 - e^{-3 \ln n} = 1 - \frac{1}{n^3}.
\end{aligned}$$

Therefore, in $T = T_1\delta_0/4 = O(n \log n)$ rounds, u will connect to at least $\delta_0/4$ new nodes with probability at least $1 - 1/n^2$, i.e., $d_T(u) \geq (1 + 1/4)\,\delta_0$. □

We next consider the second case where less than $\delta_0/3$ neighbors of a given node u are strongly tied to the two-hop neighborhood of u. This case is more challenging since the neighbors of u that are weakly tied may not contribute many new edges to u. We break the analysis of this part into two subcases based on whether there is at least one neighbor of u that is strongly tied to $N_0^2(u)$. Figure 2.1 illustrates the different cases and lemmas used in the proof of Theorem 8.

Lemma 6 (**When few neighbors are strongly tied to two-hop neighbors).** *There exists $T = O(n \log n)$ such that if less than $\delta_0/3$ nodes in $N_t^1(u)$ are strongly tied to $N_t^2(u)$ for all $t < T$, and there exists a node $v_0 \in N_0^1(u)$ that is strongly tied to $N_0^2(u)$, then $d_T(u) \geq (1 + 1/8)\,\delta_0$ with probability at least $1 - 1/n^2$.*

Proof. If at any point $t < T$, $d_T(u) \geq (1 + 1/8)\,\delta_0$, then the claim of the lemma holds. In the remainder of this proof, we assume $d_T(u) < (1 + 1/8)\,\delta_0$ for all $t < T$. Let S_t^0 denote the set of v_0's neighbors in $N_t^2(u)$ which are strongly tied to $N_t^1(u)$ at time t, W_t^0 denote the set of v_0's neighbors in $N_t^2(u)$ which are weakly tied to $N_t^1(u)$ at time t.

Consider any node v in S_t^0. Less than $\delta_0/3$ nodes in $N_t^1(u)$ are strongly tied to $N_t^2(u)$, thus more than $\delta_0/2 - \delta_0/3 = \delta_0/6$ neighbors of v in $N_t^1(u)$ are weakly tied to $N_t^2(u)$. Let w be one such weakly tied node. By Lemma 4, the probability that u connects to v through w in round t is at least $1/(4\delta_0^2)$. We have at least $\delta_0/6$ such w's, each of which executes a triangulation step each round. Consider $T = 72\delta_0 \ln n$ rounds of the process. Then the probability that u connects to v in T rounds is at least

$$1 - \left(1 - \frac{1}{4\delta_0^2}\right)^{12\delta_0^2 \ln n} \geq 1 - e^{-3\ln n} = 1 - \frac{1}{n^3}.$$

Thus, if $|S_t^0| \geq \delta_0/8$, in an additional $O(n \log n)$ time, $d_T(u) \geq (1 + 1/8)\delta_0$ with probability at least $1 - 1/n^2$.

Therefore, in the remainder of the proof we consider the case where $|S_t^0| < \delta_0/8$. Define $R_t^0 = R_{t-1}^0 \cup W_t^0$, $R_0^0 = W_0^0$. If at least $\delta_0/8$ nodes in R_t^0 are connected to u at any time, then the claim of the lemma holds. Thus, in the following we consider the case where $|R_t^0 \cap N_t^1(u)| < \delta_0/8$. From the definition of R_t^0, we can derive

$$|R_t^0| \geq |W_t^0| = d_t\left(v_0, N_t^2(u)\right) - |S_t^0| \geq d_t\left(v_0, N_t^2(u)\right) - \delta_0/8$$

At time 0, v_0 is strongly tied to $N_0^2(u)$, i.e., $d_0\left(v_0, N_0^2(u)\right) \geq \delta_0/2$. Since $\delta_0 \leq d_t(u) < (1 + 1/8)\delta_0$, we have

$$d_t\left(v_0, N_t^2(u)\right) \geq d_t\left(v_0, N_0^2(u)\right) - \delta_0/8 \geq 3\delta_0/8$$

Let e_1 denote the event $\left\{u \text{ connects to a node in } R_t^0 \setminus N_t^1(u) \text{ through } v_0 \text{ in round } t\right\}$.

$$\begin{aligned}
\Pr[e_1] &= \frac{|R_t^0 \setminus N_t^1(u)|}{d_t(v_0)} \cdot \frac{1}{d_t(v_0)} = \frac{|R_t^0| - |R_t^0 \cap N_t^1(u)|}{d_t(v_0)} \cdot \frac{1}{d_t(v_0)} \\
&\geq \frac{|R_t^0| - |R_t^0 \cap N_t^1(u)|}{d_t(v_0)} \cdot \frac{1}{n} = \frac{|R_t^0| - |R_t^0 \cap N_t^1(u)|}{\left|N_t^1(u)\right| + d_t\left(v_0, N_t^2(u)\right)} \cdot \frac{1}{n} \\
&\geq \frac{|R_t^0| - \delta_0/8}{\left|N_t^1(u)\right| + d_t\left(v_0, N_t^2(u)\right)} \cdot \frac{1}{n} \geq \frac{d_t\left(v_0, N_t^2(u)\right) - \delta_0/8 - \delta_0/8}{\left|N_t^1(u)\right| + d_t\left(v_0, N_t^2(u)\right)} \cdot \frac{1}{n} \\
&\geq \frac{3\delta_0/8 - \delta_0/8 - \delta_0/8}{\left|N_t^1(u)\right| + 3\delta_0/8} \cdot \frac{1}{n} \geq \frac{3\delta_0/8 - \delta_0/8 - \delta_0/8}{(1+1/8)\delta_0 + 3\delta_0/8} \cdot \frac{1}{n} = \frac{1}{12n}
\end{aligned}$$

Let X_1 be the number of rounds it takes for e_1 to occur. When e_1 occurs, let v_1 denote a witness for e_1. We know v_1 is in $W_{t_1}^0$ for some t_1, i.e., v_1 is strongly tied to $N_{t_1}^2(u) \cap N_{t_1}^3(u)$. If $d_t\left(v_1, N_t^2(u)\right) < 3\delta_0/8$ at any point, then $d_t(u) \geq (1+1/8)\delta_0$. Thus, in the remainder of the proof, we consider the case where $d_t\left(v_1, N_t^2(u)\right) \geq 3\delta_0/8$. Let S_t^1 (resp., W_t^1) denote the set of v_1's neighbors in $N_t^2(u)$ that are strongly tied (resp., weakly tied) to $N_t^1(u)$. If $|S_t^1| \geq \delta_0/8$, then as we did for the case $|S_t^0| \geq \delta_0/8$, we argue that in $O(n \log n)$ rounds, the degree of u is at least $(1+1/8)\delta_0$ with probability at least $1 - 1/n^2$.

Thus, in the remainder, we assume that $|S_t^1| < \delta_0/8$. Define $R_t^1 = R_{t-1}^1 \cup W_t^1$, $R_{t_1}^1 = W_{t_1}^1$. Let e_2 denote the event $\left\{u \text{ connects to a node in } R_t^0 \setminus N_t^1(u) \text{ (or } R_t^1 \setminus N_t^1(u)) \text{ through } v_0 \text{(or } v_1)\right.$ By the same calculation as for v_0, we have $\Pr[e_2] \geq 1/6n$. Similarly, we can define $e_3, X_3, e_4, X_4, \ldots, e_{\delta_0/4}, X_{\delta_0/4}$, and obtain that $\Pr[e_i] \geq i/(12n)$. The total number of rounds for u to gain $\delta_0/4$ edges is bounded by $T = \sum_i X_i$. By Lemma 2, $T \leq 36n \ln n$ with probability at least $1 - 1/n^2$, completing the proof of this lemma. □

Lemma 7 (When all neighbors are weakly tied to two-hop neighbors). *There exists $T = O(n \log n)$ such that if all nodes in $N_t^1(u)$ are weakly tied to $N_t^2(u)$ for all $t < T$, then $d_T(u) \geq \min\{(1+1/8)\delta_0, n-1\}$ with probability at least $1 - 1/n^2$.*

Proof. If at any point $t < T$, $d_t(u) \geq \min\{(1+1/8)\delta_0, n-1\}$, then the claim of this lemma holds. In the remainder of this proof, we assume $d_t(u) < \min\{(1+1/8)\delta_0, n-1\}$ for all $t < T$. In the following, we first show, any node $v \in N_0^2(u)$ will have at least $\delta_0/4$ edges to $N_{T_1}^1(u)$, where $T_1 = O(n \log n)$. After that, v will connect to u in $T_2 = O(n \log n)$ rounds. Therefore, the total number of rounds used for v to connect to u is $T_3 = T_1 + T_2 = O(n \log n)$.

Node v at least connects to one node in $N_0^1(u)$. Call it w_1. Because all nodes in $N_t^1(u)$ are weakly tied to $N_t^2(u)$, we have $d_t\left(w_1, N_t^1(u)\right) \geq \delta_0 - \delta_0/2 = \delta_0/2$. If $d_t\left(w_1, N_t^1(u) \setminus N_t^1(v)\right) < \delta_0/4$, then v already has $\delta_0/4$ edges to $N_t^1(u)$. Thus, in the following we consider the case where $d_t\left(w_1, N_t^1(u) \setminus N_t^1(v)\right) \geq \delta_0/4$. Let e_1 denote the event $\left\{v \text{ connects to a node in } N_t^1(u) \setminus N_t^1(v) \text{ through } w_1\right\}$.

$$\begin{aligned}
\Pr[e_1] &= \frac{d_t\left(w_1, N_t^1(u) \setminus N_t^1(v)\right)}{d_t(w_1)} \cdot \frac{1}{d_t(w_1)} \geq \frac{d_t\left(w_1, N_t^1(u) \setminus N_t^1(v)\right)}{\left|N_t^1(u)\right| + d_t\left(w_1, N_t^2(u)\right)} \cdot \frac{1}{d_t(w_1)} \\
&\geq \frac{\delta_0/4}{(1+1/8)\delta_0 + \delta_0/2} \cdot \frac{1}{d_t(w_1)} \geq \frac{2}{13} \cdot \frac{1}{n} > \frac{1}{7n}
\end{aligned}$$

Let X_1 be the number of rounds needed for e_1 to occur. When e_1 occurs, let w_2 denote a witness for e_1. Notice w_2 is also weakly tied to $N_t^2(u)$. By similar argument, we have $d_t\left(w_2, N_t^1(u) \setminus N_t^1(v)\right) \geq \delta_0/4$. Let e_2 denote the event $\left\{v \text{ connects to a node in } N_t^1(u) \text{ through } w_1 \text{ or } w_2\right\}$. We have $\Pr[e_2] \geq 2/(7n)$. Let X_2 be the number of rounds needed for e_2 to occur. Similarly, we can define $e_3, X_3, \ldots, e_{\delta_0/4}, X_{\delta_0/4}$ and show $\Pr[e_i] \geq i/(7n)$. Set $T_1 = \sum_i X_i$, which is the bound on the number of rounds needed for v to have at least $\delta_0/4$ neighbors in $N_t^1(u)$. By Lemma 2, we know $T_2 \leq 28n \ln n$ with probability at least $1 - 1/n^3$. Now we show v will connect to u in T_2 time after this. Notice that, all w_i's are still weakly tied to $N_t^2(u)$. By Lemma 4, the probability that u connects to v through w_i in round t is at least $1/(4\delta_0^2)$. We have $w_1, w_2, \ldots, w_{\delta_0/4}$ independently executing a triangulation step each round. Consider $T_2 = 48\delta_0 \ln n$ rounds of the process. Then,

$$\Pr[u \text{ connects to } v \text{ in } T_2 \text{ rounds}] \geq 1 - \left(1 - \frac{1}{4\delta_0^2}\right)^{12\delta_0^2 \ln n} \geq 1 - \frac{1}{n^3}.$$

Combine the two steps. We have shown for any node $v \in N_0^2(u)$, it will connect to u in time $T_3 = T_1 + T_2$ with probability at least $1 - 1/n^3$. This implies in time T_3, u will connect to all nodes in $N_0^2(u)$ with probability at least $1 - \left|N_0^2(u)\right|/n^3$. Then, $N_0^2(u) \subseteq N_{T_3}^1(u), N_0^3(u) \subseteq N_{T_3}^1(u) \cup N_{T_3}^2(u), N_0^4(u) \subseteq N_{T_3}^1(u) \cup N_{T_3}^2(u) \cup N_{T_3}^3(u)$. Now we apply the above analysis twice, and obtain that in time $T = 3T_3 = O(n \log n)$, $N_0^2(u) \cup N_0^3(u) \cup N_0^4(u) \subseteq N_T^1(u)$ with probability at least $1 - \left|N_0^2(u) \cup N_0^3(u) \cup N_0^4(u)\right|/n^3 \geq 1 - 1/n^2$. By Lemma 1, we know $\left|N_0^2(u) \cup N_0^3(u) \cup N_0^4(u)\right| \geq \min\{2\delta_0, n-1\}$. Thus, we complete the proof of this lemma. □

Theorem 8 (**Upper bound for triangulation process**). *For any connected undirected graph, the triangulation process converges to a complete graph in $O(n \log^2 n)$ rounds with high probability.*

Proof. We first show that in $O(n \log n)$ rounds, either the graph becomes complete or the minimum degree of the graph increases by a factor of at least $1/12$. Then we apply this argument $O(\log n)$ times to complete the proof of this theorem.

For each u where $d_0(u) < \min\{(1+1/8)\delta_0, n-1\}$, we consider the following 2 cases. The first case is if more than $\delta_0/3$ nodes in $N_0^1(u)$ are strongly tied to $N_0^2(u)$. By Lemma 5, there exists $T = O(n \log n)$ such that if at least $\delta_0/4$ nodes in $N_t^1(u)$ are strongly tied to $N_t^2(u)$ for $t < T$, then $d_T(u) \geq (1+1/8)\delta_0$ with probability at least $1 - 1/n^2$. Whenever the condition is not satisfied, i.e., less than $\delta_0/4$ nodes in $N_t^1(u)$ are strongly tied to $N_t^2(u)$, it means more than $\delta_0/3 - \delta_0/4 = \delta_0/12$ strongly tied nodes became weakly tied. By the definitions of strongly tied and weakly tied, this implies $d_T(u) \geq (1+1/12)\delta_0$.

The second case is if less than $\delta_0/3$ nodes in $N_0^1(u)$ are strongly tied to $N_0^2(u)$. By Lemmas 6 and 7, we know that there exists $T = O(n \log n)$ such that if we remain in this case for T rounds, then $d_T(u) \geq \min\{(1+1/8)\delta_0, n-1\}$ with probability at least $1 - 1/n^2$. Whenever the condition is not satisfied, i.e., more than $\delta_0/3$ nodes in $N_t^1(u)$ are strongly tied to $N_t^2(u)$, we move to the analysis in the first case, and $d_T(u) \geq (1+1/8)\delta_0$ in $T = O(n \log n)$ time with probability at least $1 - 1/n^2$.

Combining the above 2 cases and applying a union bound, we obtain $\delta_T \geq \min\{(1+1/8)\delta_0, n-1\}$ in $T = O(n \log n)$ rounds with probability at least $1 - 1/n$. We now apply the above argument $O(\log n)$ times to obtain the desired $O(n \log^2 n)$ upper bound. □

2.2.2 Lower bound

Theorem 9 (Lower bound for triangulation process). *For any connected undirected graph G that has $k \geq 1$ edges less than the complete graph the triangulation process takes $\Omega(n \log k)$ steps to complete with probability at least $1 - O\left(e^{-k^{1/4}}\right)$.*

Proof. We first observe that during the triangulation process there is a time t when the number of missing edges is at least $m = O(\sqrt{k})$ and the minimum degree is at least $n/3$. If $k < \frac{2}{3}n$ then this is true initially and for larger k this is true at the first time t the minimum degree is large enough. The second case follows since the degree of a node (and thus also the minimum degree) can at most double in each step guaranteeing that the minimum degree is not larger than $\frac{2}{3}n$ at time t also implying that at least $\frac{n}{3} = \Omega(\sqrt{k})$ edges are still missing.

Given the bound on the minimum degree any missing edge $\{u, v\}$ is added by a fixed node w with probability at most $\frac{9}{2n^2}$. Since there are at most $n-2$ such nodes the probability that a missing edge gets added is at most $\frac{9}{2n}$. To analyze the time needed for all missing edges to be added we denote with X_i the random variable counting the number of steps needed until the ith of the m missing edges is added. We would like to analyze $\Pr[X_1 \leq T, X_2 \leq T, \ldots, X_m \leq T]$ for an appropriately chosen number of steps

T. Note that the events $X_i < T$ and $X_j < T$ are not independent and indeed can be positively or negatively correlated. Nevertheless, independent of the conditioning onto any of the events $X_j < T$, we have that $\Pr[X_1 \leq T] \leq 1 - (1 - \frac{9}{2n})^T \leq 1 - \frac{1}{\sqrt{m}}$ for an appropriately chosen $T = \Omega(n \log m)$, where m is again the number of missing edges at time t. Thus,

$$\Pr[X_1 \leq T, X_2 \leq T, \ldots, X_m \leq T] =$$

$$= \Pr[X_1 \leq T | X_2 \leq T, \ldots, X_m \leq T] \cdot \Pr[X_2 \leq T | X_3, \ldots, X_m \leq T] \cdot \ldots \cdot \Pr[X_m \leq T]$$

$$\leq \left(1 - \frac{1}{\sqrt{m}}\right)^m = O\left(e^{-\sqrt{m}}\right) = O\left(e^{-k^{1/4}}\right)$$

This shows that the triangulation process takes with probability at least $1 - O\left(e^{-k^{1/4}}\right)$ at least $\Omega(n \log m) = O(n \log k)$ steps to complete. □

2.3 The two-hop walk: Discovery through pull

In this section, we analyze the two-hop walk process on undirected connected graphs, which is described by the following simple iteration: In each round, for each node u, we add edge (u, w) where w is drawn uniformly at random from $N_t^1(v)$, where v is drawn uniformly at random from $N_t^1(u)$. The two-hop walk yields the following pull-based resource discovery protocol. In each round, each node u contacts a random neighbor v, receives the identity of a random neighbor w of v, and sends its identity to w. The main result of this section is that the two-hop walk process transforms an arbitrary connected n-node graph to a complete graph in $O(n \log^2 n)$ rounds with high probability. We also establish an $\Omega(n \log n)$ lower bound on the two-hop walk for almost all n-node graphs.

2.3.1 Upper bound

As for the triangulation process, we establish the $O(n \log^2 n)$ upper bound by showing that the minimum degree of the graph increases by a constant factor (or equals $n - 1$) in $O(n \log n)$ rounds with high probability. For analyzing the growth in the degree of a node u, we consider two overlapping cases. The first case is when the two-hop neighborhood of u is not too large, i.e., $|N_t^2(u)| < \delta_0/2$, and the second is when the two-hop neighborhood of u is not too small, i.e., $|N_t^2(u)| \geq \delta_0/4$. As in the analysis of the triangulation process, we also use the notions of strongly and weakly tied based on how many edges connect a node to a given set; it is more convenient to work with a different threshold. We say that a node v is ***weakly tied*** to a set of nodes S if v has less than $\delta_0/4$ edges to S (i.e. $d_t(v, S) < \delta_0/4$), and ***strongly tied*** to S if v has at least $\delta_0/4$ edges to S (i.e. $d_t(v, S) \geq \delta_0/4$).

Lemma 10 (When the two-hop neighborhood is not too large). *There exists $T = O(n \log n)$ such that either $\left|N_T^2(u)\right| \geq \delta_0/2$ or $d_T(u) \geq \min\{2\delta_0, n-1\}$ with probability at least $1 - 1/n^2$.*

Proof. By the definition of δ_0, $d_0(w) \geq \delta_0$ for all w in $N_0^1(u)$. Let X be the first round at which $|N_X^2(u)| \geq \delta_0/2$. We consider two cases. If X is at most $cn \log n$ for a constant c to be specified later, then the claim of the lemma holds. In the remainder of this proof we consider the case where X is greater than $cn \log n$; thus, for $0 \leq t \leq cn \log n$, $|N_t^2(u)| < \delta_0/2$.

Consider any node w in $N_0^1(u)$. Since $d_0(w) \geq \delta_0$ and $|N_t^2(u)| < \delta_0/2$, w has at least $\delta_0/2$ edges to nodes in $N_0^1(u)$. Fix a node v in $N_0^2(u)$. In the following, we first show that in $O(n \log n)$ rounds, v is strongly tied to the neighbors of u with probability at least $1-1/n^3$. Let T_1 denote the first round at which v has is strongly tied to $N_{T_1}^1(u)$, i.e., when $|N_{T_1}^1(v) \cap N_{T_1}^1(u)| \geq \delta_0/4$. We know that v has at least one neighbor, say w_1, in $N_0^1(u)$. Consider any $t < T_1$. Since v is weakly tied to $N_0^1(u)$ at time t, w_1 has at least $\delta_0/4$ neighbors in $N_0^1(u)$ which do not have an edge to v at time t. This implies

$$\Pr\left[v \text{ connects to a node in } N_0^1(u) \text{ through } w_1 \text{ in round } t\right] \geq \frac{1}{n} \cdot \frac{1}{4} = \frac{1}{4n}$$

Let e_1 denote the event $\left\{v \text{ connects to a node in } N_0^1(u)\right\}$, and X_1 be the number of rounds for e_1 to occur. When e_1 occurs, let w_2 denote a witness for e_1. We note that $w_1, w_2 \in N_0^1(u) \subseteq N_{X_1}^1(u)$. If v is weakly tied to $N_{X_1}^1(u)$, both w_1 and w_2 have at least $\delta_0/4$ neighbors in $N_{X_1}^1(u)$ that do not have an edge to v yet. Let e_2 denote the event $\left\{v \text{ connects to a node in } N_{X_1}^1(u)\right\}$, and X_2 be the number of rounds for e_2 to occur. Then $\Pr[e_2] = 2\Pr[e_1] \geq 1/2n$. Similarly, we define $e_3, X_3, \ldots, e_{\delta_0/4}, X_{\delta_0/4}$ and obtain $\Pr[e_i] \geq i/(4n)$. We now apply Lemma 2 to obtain that $X_1 + X_2 + \ldots X_{\delta_0/4}$ is at most $16n \ln n$ with probability at least $1 - 1/n^3$. Thus, with probability at least $1 - |N_0^2(u)|/n^3$, $T_1 \leq 16n \ln n$. After T_1 rounds, we obtain that for any $v \in N_0^2(u)$,

$$\Pr[u \text{ connects to } v \text{ in a single round}] \geq \frac{\delta_0/4}{2\delta_0} \cdot \frac{1}{n} = \frac{1}{8n}.$$

which implies that with probability at least $1 - 1/n^3$, u has an edge to every node in $N_0^2(u)$ in another $T_2 \leq 24n \ln n$ rounds.

Let T_3 equal $T_1 + T_2$; we set c to be at least $120 \ln 2$ so that $X > 3T_3$. We thus have $N_0^2(u) \subseteq N_{T_3}^1(u)$, $N_0^3(u) \subseteq N_{T_3}^1(u) \cup N_{T_3}^2(u)$, and $N_0^4(u) \subseteq N_{T_3}^1(u) \cup N_{T_3}^2(u) \cup N_{T_3}^3(u)$. We now repeat the above analysis again twice and obtain that at time $T = 3T_3$, $N_0^2(u) \cup N_0^3(u) \cup N_0^4(u) \subseteq N_T^1(u)$ with probability at least $1 - \left|N_0^2(u) \cup N_0^3(u) \cup N_0^4(u)\right| / n^3 \geq 1 - 1/n^2$. By Lemma 1, we have $\left|N_T^1(u)\right| \geq \min\{2\delta_0, n-1\}$, thus completing the proof of the lemma. □

Lemma 11 (When the two-hop neighborhood is not too small). *There exists $T = O(n \log n)$ such that either $\left|N_T^2(u)\right|$ is less than $\delta_0/4$ or $d_T(u)$ is at least $\min\{(1+1/8)\delta_0, n-1\}$, with probability at least $1 - 1/n^2$.*

Proof. Let X be the first round at which $N_X^2(u) < \delta_0/4$. We consider two cases. If X is at most $cn \log n$ for a constant c to be specified later, then the claim of the lemma holds. In the remainder of this proof we consider the case where X is greater than $cn \log n$; thus, for $0 \leq t \leq cn \log n$, $\left|N_t^2(u)\right| \geq \delta_0/4$. If $v \in N_0^2(u)$ is strongly tied to $N_0^1(u)$, then

$$\Pr[u \text{ connects to } v \text{ in a single round}] \geq \frac{d_t\left(v, N_0^1(u)\right)}{\left|N_t^1(u)\right|} \cdot \frac{1}{n} \geq \frac{\delta_0/4}{(1+1/8)\delta_0} \cdot \frac{1}{n} = \frac{2}{9n}$$

Thus, in $T = 13.5 n \ln n$ rounds, u will add an edge to v with probability at least $1 - 1/n^3$. If there are at least $\delta_0/8$ nodes in $N_0^2(u)$ that are strongly tied to $N_0^1(u)$, then u will add edges to all these nodes in T rounds with probability at least $1 - 1/n^2$.

In the remainder of this proof, we focus on the case where the number of nodes in $N_0^2(u)$ that are strongly tied to $N_0^1(u)$ at the start of round 0 is less than $\delta_0/8$. In this case, because $\left|N_t^2(u)\right| \geq \delta_0/4$, more than $\delta_0/8$ nodes in $N_0^2(u)$ are weakly tied to $N_0^1(u)$, and, thus, have at least $3\delta_0/4$ edges to nodes in $N_0^2(u) \cup N_0^3(u)$.

In the following we show u will connect to $\delta_0/8$ nodes in $O(n \log n)$ rounds with probability at least $1 - 1/n^2$. For any round t, let W_t denote the set of nodes in $N_t^2(u)$ that are weakly tied to $N_t^1(u)$. We refer to a length-2 path from u to a node two hops away as an *out-path*. Let P_0 denote the set of out-paths to W_0. Since we have at least $\delta_0/8$ nodes in $N_0^2(u)$ that are weakly tied to $N_0^1(u)$, $|P_0|$ is at least $\delta_0/8$ at time $t = 0$. Define $e_1 = \left\{u \text{ picks an out-path in } P_0 \text{ and connects to node } v_1 \text{ in } N_0^2(u)\right\}$, and X_1 to be the number of rounds for e_1 to occur. When $0 \leq t \leq X_1$, for each $w_i \in N_t^1(u)$, let f_i be the number of edges from w_i to nodes in $N_t^1(u) \cup N_t^2(u)$, and p_i be the number of edges from w_i to nodes in $N_0^2(u)$ that are weakly tied to $N_0^1(u)$.

$$\begin{aligned}\Pr[e_1] &= \sum_i \frac{1}{d_t(u)} \cdot \frac{p_i}{f_i} \geq \sum_i \frac{1}{d_t(u)} \cdot \frac{p_i}{n-1} = \frac{\sum_i p_i}{(1+1/8)\delta_0(n-1)} \\ &= \frac{|S|}{(1+1/8)\delta_0(n-1)} \geq \frac{\delta_0/8}{(1+1/8)\delta_0(n-1)} \geq \frac{1}{9n}.\end{aligned}$$

After X_1 rounds, u will pick an out-path in P_0 and connect such a v_1. Define P_1 to be a set of out-paths from u to W_{X_1}. We now place a lower bound on $|P_1 \setminus P_0|$. Since $v_1 \in N_0^2(u)$ is added to $N_{X_1}^1(u)$, those out-paths in P_0 consisting of edges from v_1 to nodes in $N_0^1(u)$ are not in P_1. The number of out-paths we lose because of this is at most $\delta_0/4$. But v_1 also has at least $3\delta_0/4$ edges to $N_0^2(u) \cup N_0^3(u)$. The end points of these edges are in $N_{X_1}^1(u) \cup N_{X_1}^2(u)$. If more than $\delta_0/8$ of them are in $N_{X_1}^1(u)$, then $d_{X_1}(u) \geq (1+1/8)\delta_0$. Now let's consider the case that less than $\delta_0/8$ such end points are in $N_{X_1}^1(u)$. This means the number of edges from v_1 to $N_{X_1}^2(u)$ is at least $3\delta_0/4 - \delta_0/4 - \delta_0/8 = 3\delta_0/8$. Among the end points of these edges, if more than $\delta_0/8$ of them are strongly tied to $N_{X_1}^1(u)$, then the degree of u will become at least $(1+1/8)\delta_0$ in $O(n \log n)$ rounds with probability $1 - 1/n^2$ by our earlier argument. If not, we know that more than $\delta_0/4$ newly added edges are pointing to nodes that are weakly tied to $N_{X_1}^1(u)$. Thus, $|P_1 \setminus P_0|$ is by at least $\delta_0/4$. $|S| \geq 2 \cdot \delta_0/8$. Define $e_2 = \{u$ picks an out-path in P_1 and connects to node $v_2\}$, and X_2 to be the number of rounds for e_2 to occur. During time $X_1 \leq t \leq X_2$, $\Pr[e_2]$ is at least $2 \cdot \frac{1}{9n}$. Similarly, we define $e_3, X_3, \ldots, e_{\delta_0/8}, X_{\delta_0/8}$ and derive $\Pr[e_i] \geq i/(9n)$. By Lemma 2, the number of rounds for $d_t(u) \geq (1+1/8)\delta_0$ is bounded by

$$T = X_1 + X_2 + \cdots + X_{\delta_0/8} \leq (2+1)9n \ln n = 27n \ln n$$

with probability at least $1 - 1/n^2$, completing the proof of this lemma. □

Theorem 12 (Upper bound for two-hop walk process). *For connected undirected graphs, the two-hop walk process completes in $O(n \log^2 n)$ rounds with high probability.*

Proof. We first show that in time $T = O(n \log n)$ time, the minimum degree of the graph increases by a factor of 1/8, i.e., $\delta_T \geq \min\{(1+1/8)\delta_0, n-1\}$. Then we can apply this argument $O(\log n)$ times, and thus, complete the proof of this theorem.

For each u where $d_0(u) < \min\{(1+1/8)\delta_0, n-1\}$, we analyze by the following 2 cases. First, if $|N_0^2(u)| \geq \delta_0/2$, by Lemma 11 we know as long as $|N_t^2(u)| \geq \delta_0/4$ for all $t \geq 0$, $d_T(u) \geq \min\{(1+1/8)\delta_0, n-1\}$ with probability $1-1/n^2$ where $T = O(n \log n)$. Whenever the condition is not satisfied, we know at least $\delta_0/4$ nodes in $N_0^2(u)$ has been moved to $N_T^1(u)$, which means $d_T(u) \geq \min\{(1+1/4)\delta_0, n-1\}$.

Second, if $|N_0^2(u)| < \delta_0/2$, by Lemma 10 we know as long as $|N_t^2(u)| < \delta_0/2$ for all $t \geq 0$, $d_T(u) \geq \min\{(1+1/8)\delta_0, n-1\}$ with probability $1-1/n^2$ where $T = O(n \log n)$. Whenever the condition is not satisfied, we are back to the analysis in the first case, and the minimum degree will become $\min\{(1+1/8)\delta_0, n-1\}$ with probability $1-1/n^2$.

Combine the above 2 cases, since we at most have n nodes whose degree is between δ_0 and $\min\{(1+1/8)\delta_0, n-1\}$, the minimum degree of G will become at least $\min\{(1+1/8)\delta_0, n-1\}$ in $O(n \log n)$ rounds with probability $1-1/n$.

Now we can apply the above argument $O(\log n)$ times, and have shown the two-hop walk process completes in $O(n \log^2 n)$ with high probability. □

2.3.2 Lower bound

Theorem 13 (Lower bound for two-hop walk process). *For any connected undirected graph G that has $k \geq 1$ edges less than the complete graph the two-hop process takes $\Omega(n \log k)$ steps to complete with probability at least $1 - O\left(e^{-k^{1/4}}\right)$.*

The proof of Theorem 13 is essentially the same as Theorem 9, and is omitted here.

2.4 Two-hop walk in directed graphs

In this section, we analyze the two-hop walk process in directed graphs. We say that the process terminates at time t if for every node u and every node v, G_t contains the edge (u, v) whenever u has a path to v in G_0.

Theorem 14. *On any n-node directed graph, the two-hop walk terminates in $O(n^2 \log n)$ rounds with high probability. Furthermore, there exists a (weakly connected) directed graph for which the process takes $\Omega(n^2 \log n)$ rounds to terminate.*

Proof. Consider any pair of nodes, u and v. Consider a shortest path from u to v $(v_0, v_1, v_2, \ldots, v_m)$, where $v_0 = u$, $v_m = v$ and $m \leq n$. Fix a time step t. Let e_i denote the event an edge is added from v_i to v_{i+2} in step t. The probability of occurrence of e_i is $\Pr[e_i] \geq 1/n^2$. All the e_i's are independent from one another.

$$\begin{aligned}
\Pr[\cup_i e_i] &\geq \sum_i \Pr[e_i] - \sum_i \sum_j \Pr[e_i \cap e_j] \\
&= \sum_i \Pr[e_i] - \sum_i \sum_j \Pr[e_i] \Pr[e_j] \\
&\geq m\frac{1}{n^2} - m(m-1)\frac{1}{n^4} \\
&\geq \frac{m}{n^2}
\end{aligned}$$

Let X_1 denote the number of steps it takes for the length of the above path to decrease by 1. It is clear that $E[X_1] \leq n^2/m$. In general, let X_i denote the number of steps it takes for the length of the above path to decrease by i. By Lemma 2, the number of steps it takes for the above path to shrink to an edge is at most $4n^2 \ln n$ with probability $1/n^3$. Taking a union bound over all the edges yields the desired upper bound.

For the lower bound, consider a graph G_0 with the node set $\{1, 2, \ldots, n\}$ and the edge set

$$\{(3i, j), (3i{+}1, j) : 0 \leq i < n/4, 3n/4 \leq j < n\} \bigcup \{(3i, 3i{+}1), (3i{+}1, 3i{+}2) : 0 \leq i < n/4\}.$$

The only edges that need to be added by the two-hop process are the edges $(3i, 3i+2)$ for $0 \leq i < n/4$. The probability that node $3i$ adds the edge $(3i, 3i+2)$ in any round is at most $16/n^2$. The probability that edge $(3i, 3i+2)$ is not added in $(n^2 \ln n)/32$ rounds is at least $1/\sqrt{n}$. Since the events associated with adding each of these edges are independent, the probability that all the $n/3$ edges are added in $(n^2 \ln n)/32$ rounds is at most $(1 - 1/\sqrt{n})^{n/3} \leq e^{-\sqrt{n}/3}$, completing the lower bound proof. □

The lower bound in the above theorem takes advantage of the fact that the initial graph is not strongly connected. Extending the above analysis for strongly connected graphs appears to be much more difficult since the events corresponding to the addition of new edges interact in significant ways. We present an $\Omega(n^2)$ lower bound for a strongly connected graph by a careful analysis that tracks the event probabilities with time and takes dependencies into account.

Theorem 15. *There exists a strongly connected directed graph G_0 for which the expected number of rounds taken by the two-hop process is $\Omega(n^2)$.*

Proof. The graph $G_0 = (V, E)$ is depicted in Figure 2.2 and formally defined as $G_0 = (V, E)$ where $V = \{1, 2, \ldots, n\}$ with n being even, and

$$E = \{(i, j) : 1 \leq i, j \leq n/2\} \cup \{(i, i+1) : n/2 \leq i < n\} \cup \{(i, j) : i > j, i > n/2, i, j \in V\}.$$

We first establish an upper bound on the probability that edge $(i, i+h)$ is added by the start of round t, for given i, $1 \leq i \leq n-h$. Let $p_{h,t}$ denote this probability. The following base cases are immediate: $p_{h,0}$ is 1 for $h = 1$ and $h < 0$, and 0 otherwise. Next, the edge $(i, i+h)$ is in G_{t+1} if and only if $(i, i+h)$ is either in G_{t-1} or added in round t. In the latter case, $(i, i+h)$ is added by a two-hop walk $i \to i+k \to i+h$,

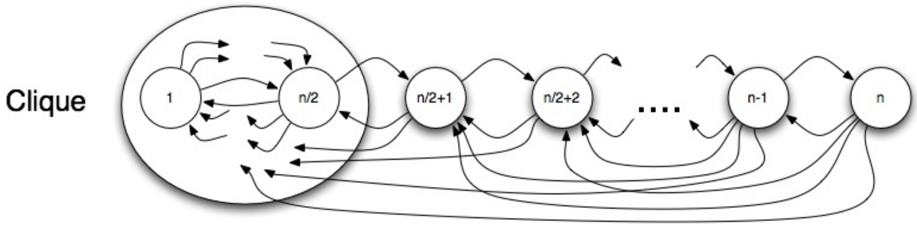


Figure 2.2: Lower bound example for two-hop walk process in directed graphs

where $-i < k \leq n - i$. Since the out-degree of every node is at least $n/2$, for any k the probability that i takes such a walk is at most $4/n^2$.

$$\begin{aligned}
p_{h,t+1} &\leq p_{h,t} + \frac{4}{n^2} \sum_{k>-i}^{n-i} p_{k,t} p_{h-k,t} \\
&= p_{h,t} + \frac{4}{n^2} \left(\sum_{k=1}^{i-1} p_{h+k,t} + \sum_{k=1}^{h-1} p_{k,t} p_{h-k,t} + \sum_{k=h+1}^{n-i} p_{k,t} \right) \qquad (2.1)
\end{aligned}$$

We show by induction on t that

$$p_{h,t} \leq \left(\frac{\alpha t}{n^2} \right)^{h-1}, \text{ for all } t \leq \epsilon n^2 \qquad (2.2)$$

where α and ϵ are positive constants that are specified later.

The induction base is immediate. For the induction step, we use the induction hypothesis for t and Equation 2.1 and bound $p_{h,t+1}$ as follows.

$$\begin{aligned}
p_{h,t+1} &\leq \left(\frac{\alpha t}{n^2} \right)^{h-1} + \frac{4}{n^2} \left(\sum_{k=1}^{i-1} \left(\frac{\alpha t}{n^2} \right)^{h+k-1} + \sum_{k=1}^{h-1} \left(\frac{\alpha t}{n^2} \right)^{k-1} \left(\frac{\alpha t}{n^2} \right)^{h-k-1} + \sum_{k=h+1}^{n-i} \left(\frac{\alpha t}{n^2} \right)^{k-1} \right) \\
&\leq \left(\frac{\alpha t}{n^2} \right)^{h-1} + \frac{4}{n^2} \left((h-1) \left(\frac{\alpha t}{n^2} \right)^{h-2} + \left(\frac{\alpha t}{n^2} \right)^{h} \frac{2}{1 - \alpha t/n^2} \right) \\
&\leq \left(\frac{\alpha t}{n^2} \right)^{h-1} + (h-1) \left(\frac{\alpha t}{n^2} \right)^{h-2} \frac{1}{n^2} \left(4 + \frac{4\epsilon^2}{(1-\alpha\epsilon)} \right) \\
&\leq \left(\frac{\alpha t}{n^2} \right)^{h-1} + (h-1) \left(\frac{\alpha t}{n^2} \right)^{h-2} \frac{\alpha}{n^2} \\
&\leq \left(\frac{\alpha(t+1)}{n^2} \right)^{h-1}.
\end{aligned}$$

(In the second inequality, we combine the first and third summations and bound them by their infinite sums. In the third inequality, we use $t \leq \epsilon n^2$. For the fourth inequality, we set α sufficiently large so that $\alpha \geq 4 + 4/(1 - \alpha\epsilon)$. The final inequality follows from Taylor series expansion.)

For an integer x, let C_x denote the cut $(\{u : u \leq x\}, \{v, v > x\})$. We say that a cut C_x is *untouched* at the start of round t if the only edge in G_t crossing the cut C_x is the edge $(x, x+1)$; otherwise, we say C_x is *touched.* Let X denote the smallest integer such that C_X is untouched. We note that X is a random variable that also varies with time. Initially, $X = n/2$.

We divide the analysis into several phases, numbered from 0. A phase ends when X changes. Let X_i denote the value of X at the start of phase i; thus $X_0 = n/2$. Let T_i denote the number of rounds in phase i. A new edge is added to the cut C_{X_i} only if either X_i selects edge $(X_i, X_i + 1)$ as its first hop or a node $u < X_i$ selects $u \to X_i \to X_i + 1$. Since the degree of every node is at least $n/2$, the probability that a new edge is added to the cut C_i is at most $2/n + n(4/n^2) = 6/n$, implying that $E[T_i] \geq n/6$.

We now place a bound on X_{i+1}. Fix a round $t \leq \epsilon n^2$, and let E_x denote the event that C_x is touched by round t. We first place an upper bound on the probability of E_x for arbitrary x using Equation 2.2.

$$\Pr[E_x] \leq \sum_{h \geq 2} h \left(\frac{\alpha t}{n^2}\right)^{h-1} \leq \frac{\alpha t(4 - 3(\alpha t)/n^2 + (\alpha t)^2/n^4)}{n^2(1 - (\alpha t)/n^2)^3},$$

for $t \leq \epsilon n^2$, where we use the inequality $\sum_{h \geq 2} h^2 \delta^h = \delta(4 - 3\delta + \delta^2)/(1 - \delta)^3$ for $0 < \delta < 1$. We set ϵ sufficiently small so that $(4 - 3\epsilon + \epsilon^2)/(1 - \epsilon)^3 \leq 5$, implying that the above probability is at most 5ϵ.

If E_x were independent from E_y for $x \neq y$, then we can invoke a straightforward analysis using a geometric probability distribution to argue that $E[X_{i+1} - X_i]$ is at most $1/(1 - 5\epsilon) = O(1)$. The preceding independence does not hold, however; in fact, for $y > x$, $\Pr[E_y \bmod E_x] > \Pr[E_y]$. We show that the impact of this correlation is very small when x and y are sufficiently far apart. We consider a sequence of cuts $C_{x_1}, C_{x_2}, \ldots, C_{x_\ell}, \ldots$ where $x_0 = X_i + 2$ and $x_\ell = x_{\ell-1} + c\ell$, for a constant c chosen sufficiently large. We bound the conditional probability of E_{x_ℓ} given $E_{x_{\ell-1}} \cap E_{x_{\ell-2}} \cdots E_{x_1}$

as follows.

$$\begin{aligned}
& \Pr[E_{x_\ell} | E_{x_{\ell-1}} \cap E_{x_{\ell-2}} \cdots E_{x_1}] \\
= \; & \frac{\Pr[E_{x_\ell} \cap E_{x_{\ell-1}} \cap E_{x_{\ell-2}} \cdots E_{x_1}]}{\Pr[E_{x_{\ell-1}} \cap E_{x_{\ell-2}} \cdots E_{x_1}]} \\
\leq \; & \frac{\Pr[E_{x_{\ell-1}} \cap E_{x_{\ell-2}} \cdots E_{x_1} \cap (C_{x_\ell} \cap (C_{x_{\ell-1}} \cup \cdots \cup C_{x_1}) = \emptyset)]}{\Pr[E_{x_{\ell-1}} \cap E_{x_{\ell-2}} \cdots E_{x_1}]} + \\
& \frac{\Pr[E_{x_{\ell-1}} \cap E_{x_{\ell-2}} \cdots E_{x_1} \cap (C_{x_\ell} \cap (C_{x_{\ell-1}} \cup \cdots \cup C_{x_1}) \neq \emptyset)}{\Pr[E_{x_{\ell-1}} \cap E_{x_{\ell-2}} \cdots E_{x_1}]} \\
\leq \; & \frac{\Pr[E_{x_{\ell-1}} \cap E_{x_{\ell-2}} \cdots E_{x_1}] \Pr[\text{a new edge is added from } (x_{\ell-1}+1, x_\ell) \text{ to } (x_\ell+1, n]]}{\Pr[E_{x_{\ell-1}} \cap E_{x_{\ell-2}} \cdots E_{x_1}]} \\
& \frac{\Pr[\text{an edge spanning at least } c\ell \text{ hops is added across } C_{x_\ell}]}{\Pr[E_{x_{\ell-1}} \cap E_{x_{\ell-2}} \cdots E_{x_1}]} \\
\leq \; & \Pr[E_{x_\ell}] + \frac{((\alpha t)/n^2)^{c\ell-1}}{(1-\alpha t/n^2)^2 (t/n^2)^\ell} \\
\leq \; & 5\epsilon + \epsilon = 6\epsilon,
\end{aligned}$$

where we set c sufficiently large in the last step. Since X_{i+1} is at most the smallest x_ℓ such that C_{x_ℓ} is untouched, we obtain that

$$E[X_{i+1} - X_i] \leq 2 + \sum_{\ell \geq 2} (6\epsilon)^\ell c\ell^2 = O(1).$$

We thus obtain that after $\epsilon' n$ phases, $E[X]$ is $O(n)$, where ϵ' is chosen sufficiently small so that $n - E[X]$ is $\Omega(n)$. Since the expected length of each phase is at least $n/6$, it follows that the expected number of rounds it takes for the two-hop process to complete is $\Omega(n^2)$ rounds.

□

2.5 Conclusion

We have analyzed two natural gossip-based discovery processes in networks and showed almost-tight bounds on their convergence in arbitrary networks. Our processes are motivated by the resource discovery problem in distributed networks as well as by the evolution of social networks. We would like to study variants of the processes that take into account failures associated with forming connections, the joining and leaving of nodes, or having only only a subset of nodes to participate in forming connections. We believe our techniques can be extended to analyze such situations as well. From a technical standpoint, the main problem left open by our work is to resolve the logarithmic factor gap between the upper and lower bounds. It is not hard to show that from the perspective of increasing the minimum degree by a constant factor, our analysis is tight up to constant factors. It is conceivable, however, that a sharper upper bound can be obtained by an alternative analysis that uses a “smoother” measure of progress.

Chapter 3

Diffusion under adversarial dynamics

In Chapter 2, we study diffusion under organic dynamics, where the network is altered by the diffusion process itself. In this chapter, we study similar problems, but under adversarial dynamics.

We study the fundamental problem of diffusion (also known as information spreading or gossip) in dynamic networks. In gossip, or more generally, k-gossip, there are k pieces of information (or tokens) that are initially present in some nodes and the problem is to disseminate the k tokens to all nodes. The goal is to accomplish the task in as few rounds of distributed computation as possible. It's not hard to show an $O(n + k)$ upper bound if the network is static (e.g. using delay sequence argument). However, the problem is especially challenging in dynamic networks where the network topology can change from round to round and can be controlled by an on-line adversary.

The focus of this chapter is on the power of token-forwarding algorithms, which do not manipulate tokens in any way other than storing and forwarding them. We first consider a worst-case adversarial model first studied by Kuhn, Lynch, and Oshman [89] in which the communication links for each round are chosen by an adversary, and nodes do not know who their neighbors for the current round are before they broadcast their messages. Our main result is an $\Omega(nk/\log n)$ lower bound on the number of rounds needed for any deterministic token-forwarding algorithm to solve k-gossip. This resolves an open problem raised in [89], improving their lower bound of $\Omega(n \log k)$, and matching their upper bound of $O(nk)$ to within a logarithmic factor. Our lower bound also extends to randomized algorithms against an adversary that knows in each round the outcomes of the random coin tosses in that round. Our result shows that one cannot obtain significantly efficient (i.e., subquadratic) token-forwarding algorithms for gossip in the adversarial model of [89]. We next show that token-forwarding algorithms can achieve subquadratic time in the offline version of the problem, where the adversary has to commit all the topology changes in advance at the beginning of the computation. We present two polynomial-time offline token-forwarding algorithms to solve k-gossip: (1) an $O(\min\{nk, n\sqrt{k \log n}\})$ round algorithm, and (2) an $(O(n^\epsilon), \log n)$ bicriteria approximation algorithm, for any $\epsilon > 0$, which means that if L is the number of rounds needed by an optimal algorithm, then our approximation algorithm will complete in $O(n^\epsilon L)$ rounds and the number of tokens transmitted on any edge is $O(\log n)$ in each round. Our results are a step towards understanding the power and limitation of token-forwarding algorithms in dynamic networks.

In Section 3.1 we formally define the k-gossip problem and the online/offline models we considered. Related work is in Section 3.2. We show the $\Omega(nk/\log n)$ lower bound in Section 3.3, and present our algorithms in Section 3.4. Finally, we conclude and give open problems in Section 3.5.

3.1 Model and problem statement

In this section, we formally define the k-gossip problem, the online and offline models, and token-forwarding algorithms.

The k-gossip problem. In this problem, k different tokens are assigned to a set V of $n \geq k$ nodes, where each node may have any subset of the tokens, and the goal is to disseminate all the k tokens to all the nodes.

The online model. Our online model is the worst-case adversarial model of [89]. Nodes communicate with each other using anonymous broadcast. We assume a synchronized communication. At the beginning of round r, each node in V decides what message to broadcast based on its internal state and coin tosses (for a randomized algorithm); the adversary chooses the set of edges that forms the communication network G_r over V for round r. We adopt a *strong adversary* model in which adversary knows the outcomes of the random coin tosses used by the algorithm in round r at the time of constructing G_r but is unaware at this time of the outcomes of any randomness used by the algorithm in future rounds. The only constraint on G_r is that it be connected; this is the same as the 1-interval connectivity model of [89].

As observed in [89], the above model is equivalent to the adversary knowing the messages to be sent in round r before choosing the edges for round r. We do not place any bound on the size of the messages, but require for our lower bound that each message contains at most one token. Finally, we note that under the strong adversary model, there is a distinction between randomized algorithms and deterministic algorithms since a randomized algorithm may be able to exploit the fact that in any round r, while the adversary is aware of the randomness used in that round, it does not know the outcomes of any randomness used in subsequent rounds.

The offline model. In the offline model, we are given a sequence of networks $\langle G_r \rangle$ where G_r is a connected communication network for round r. As in the online model, we assume that in each round at most one token is broadcast by any node. It can be easily seen that the k-gossip problem can be solved in nk rounds in the offline model; so we may assume that the given sequence of networks is of length at most nk.

Token-forwarding algorithms. Informally, a token-forwarding algorithm is one that does not combine or alter tokens, only stores and forwards them. Formally, we call an algorithm for k-gossip a token-forwarding algorithm if for every node v, token t, and round r, v contains t at the start of round r of the algorithm if and only if either v has t at the start of the algorithm or v received a message containing t prior to round r.

Finally, several of our arguments are probabilistic. We use the term "with high probability" to mean with probability at least $1 - 1/n^c$, for a constant c that can be made sufficiently high by adjusting related constant parameters.

3.2 Related work

Information spreading (or dissemination) in networks is one of the most basic problems in computing and has a rich literature. The problem is generally well-understood on static networks, both for interconnection networks [93] as well as general networks [96, 17]. In particular, the k-gossip problem can be solved in $O(n+k)$ rounds on any n-static network [122]. There also have been several papers on broadcasting, multicasting, and related problems in static heterogeneous and wireless networks (e.g., see [12, 26, 25, 50]).

Dynamic networks have been studied extensively over the past three decades. Some of the early studies focused on dynamics that arise out of faults, i.e., when edges or nodes fail. A number of fault models, varying according to extent and nature (e.g., probabilistic vs. worst-case) and the resulting dynamic networks have been analyzed (e.g., see [17, 96]). There have been several studies on models that constrain the rate at which changes occur, or assume that the network eventually stabilizes (e.g., see [7, 57, 66]).

There also has been considerable work on general dynamic networks. Some of the earliest studies in this area include [8, 23] which introduce general building blocks for communication protocols on dynamic networks. Another notable work is the local balancing approach of [22] for solving routing and multicommodity flow problems on dynamic networks. Algorithms based on the local balancing approach continually balance the packet queues across each edge of the network and drain packets that have reached their destination. The local balancing approach has been applied to achieve near-optimal throughput for multicast, anycast, and broadcast problems on dynamic networks as well as for mobile ad hoc networks [21, 24, 77].

Modeling general dynamic networks has gained renewed attention with the recent advent of heterogeneous networks composed out of ad hoc, and mobile devices. To address the unpredictable and often unknown nature of network dynamics, [89] introduce a model in which the communication graph can change completely from one round to another, with the only constraint being that the network is connected at each round. The model of [89] allows for a much stronger adversary than the ones considered in past work on general dynamic networks [22, 21, 24]. In addition to results on the k-gossip problem that we have discussed earlier, [89] consider the related problem of counting, and generalize their results to the T-interval connectivity model, which includes an additional constraint that any interval of T rounds has a stable connected spanning subgraph. The survey of [90] summarizes recent work on dynamic networks.

We note that the model of [89], as well as ours, allow only edge changes from round to round while the nodes remain fixed. Recently, the work of [18] introduced a dynamic network model (motivated by P2P networks) where both nodes and edges can change by a large amount (up to a linear fraction of the network size). They show that stable amost-everywhere agreement can be efficiently solved in such networks even in adversarial dynamic settings.

Recent work of [72, 73] presents information spreading algorithms based on network coding [10]. As mentioned earlier, one of their important results is that the k-gossip problem on the adversarial model of [89] can be solved using network coding in $O(n+k)$ rounds assuming the token sizes are sufficiently large ($\Omega(n \log n)$ bits). For further references to using network coding for gossip and related problems, we refer to the recent works of [72, 73, 19, 42, 53, 106] and the references therein.

Our offline approximation algorithm makes use of results on the Steiner tree packing problem for directed graphs [48]. This problem is closely related to the directed Steiner tree problem (a major open problem in approximation algorithms) [46, 130] and the gap between network coding and flow-based solutions for multicast in arbitrary directed networks [9, 118].

Finally, we note that there are also a number of studies that solve k-gossip and related problems using *gossip-based* processes. In a local gossip-based algorithm, each node exchanges information with a small number of randomly chosen neighbors in each round. Gossip-based processes have recently received significant attention because of their simplicity of implementation, scalability to large network size, and their use in aggregate computations, e.g., [34, 54, 82, 47, 80, 106, 43] and the references therein. All these studies assume an underlying static communication network, and do not apply directly to the models considered in this paper. A related recent work on dynamic networks is [20] which analyzes the cover time of random walks on dynamic networks.

3.3 Lower bound for online token-forwarding algorithms

In this section, we give an $\Omega(kn/\log n)$ lower bound on the number of rounds needed by any online token-forwarding algorithm for the k-gossip problem against a strong adversary. As discussed earlier, this immediately implies the same lower bound for any deterministic online token-forwarding algorithm. Our lower bound applies to even centralized algorithms and a large class of initial token distributions. We first describe the adversary strategy.

Adversary: The strategy of the adversary is simple. We use the notion of *free edge* introduced in [89]. In a given round r, we call an edge (u, v) to be a free edge if at the start of round r, u has the token that v broadcasts in the round and v has the token that

u broadcasts in the round[1]; an edge that is not free is called *non-free*. Thus, if (u, v) is a free edge in a particular round, neither u nor v can gain any new token through this edge in the round. Since we are considering a strong adversary model, at the start of each round, the adversary knows for each node v, the token (if any) that v will broadcast in that round. In round r, the adversary constructs the communication graph G_r as follows. First, the adversary adds all the free edges to G_r. Let $C_1, C_2, \ldots, C_l$ denote the connected components thus formed. The adversary then guarantees the connectivity of the graph by selecting an arbitrary node in each connected component and connecting them in a line. Figure 3.1 illustrates the construction.

The network G_r thus constructed has exactly $l - 1$ non-free edges, where l is the number of connected components formed by the free edges of G_r. If (u, v) is a non-free edge in G_r, then u, v, or both will gain at most new token through this edge. We refer to such a token exchange on a non-free edge as a *useful token exchange*.

We bound the running-time of any token-forwarding algorithm by identifying a critical structure that quantifies the progress made in each round. We say that a sequence of nodes $v_1, v_2, \ldots, v_k$ is *half-empty* in round r with respect to a sequence of tokens $t_1, t_2, \ldots, t_k$ if the following condition holds at the start of round r: for all $1 \leq i, j \leq k$, $i \neq j$, either v_i is missing t_j or v_j is missing t_i. We then say that $\langle v_i \rangle$ is half-empty with respect to $\langle t_i \rangle$ and refer to the pair $(\langle v_i \rangle, \langle t_i \rangle)$ as a half-empty configuration of size k.

Lemma 16. *If m useful token exchanges occur in round r, then there exists a half-empty configuration of size at least $m/2 + 1$ at the start of round r.*

Proof. Consider the network G_r in round r. Each non-free edge can contribute at most 2 useful token exchanges. Thus, there are at least $m/2$ non-free edges in the communication graph. Based on the adversary we consider, no useful token exchange takes place within the connected components induced by the free edges. Useful token exchanges can only happen over the non-free edges between connected components. This implies there are at least $m/2 + 1$ connected components in the subgraph of G_r induced by the free edges. Let v_i denote an arbitrary node in the ith connected component in this subgraph, and let t_i be the token broadcast by v_i in round r. For

[1] For convenience, when a node does not broadcast any token we will view it as broadcasting a special *empty* token that every node has. This allows us to avoid treating the empty broadcast as a special case.

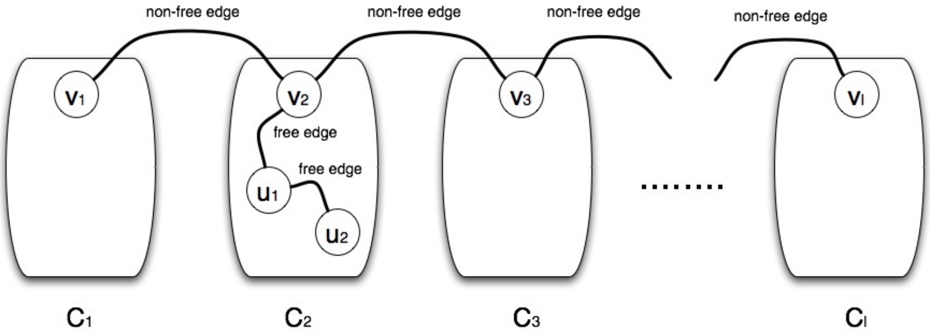


Figure 3.1: The network constructed by the adversary in a particular round. Note that if node v_i broadcasts token t_i, then the $\langle v_i \rangle$ forms a half-empty configuration with respect to $\langle t_i \rangle$ at the start of this round.

$i \neq j$, since v_i and v_j are in different connected components, (v_i, v_j) is a non-free edge in round r; hence, at the start of round r, either v_i is missing t_j or v_j is missing t_i. Thus, the sequence $\langle v_i \rangle$ of nodes of size at least $m/2 + 1$ is half-empty with respect to the sequence $\langle t_i \rangle$ at the start of round r. □

An important point to note about the definition of a half-empty configuration is that it only depends on the token distribution; it is independent of the broadcast in any round. This allows us to prove the following easy lemma.

Lemma 17. *If a sequence $\langle v_i \rangle$ of nodes is half-empty with respect to $\langle t_i \rangle$ at the start of round r, then $\langle v_i \rangle$ is half-empty with respect to $\langle t_i \rangle$ at the start of round r' for any $r' \leq r$.*

Proof. The lemma follows immediately from the fact that if a node v_i is missing a token t_j at the start of round r, then v_i is missing token t_j at the start of every round $r' < r$. □

Lemmas 16 and 17 suggest that if we can identify a token distribution in which all half-empty configuration are small, we can guarantee small progress in each round. We now show that there are many token distributions with this property, thus yielding the desired lower bound.

Theorem 18. *From an initial token distribution in which each node has each token independently with probability $3/4$, any online token-forwarding algorithm will need $\Omega(kn/\log n)$ rounds to complete with high probability against a strong adversary.*

Proof. We first note that if the number of tokens k is less than $100 \log n$, then the $\Omega(kn/\log n)$ lower bound is trivially true because even to disseminate one token it will take $\Omega(n)$ rounds in the worst-case. Thus, in the following proof, we focus on the case where $k \geq 100 \log n$.

Let E_l denote the event that there exists a half-empty configuration of size l at the start of the first round. For E_l to hold, we need l nodes $v_1, v_2, \ldots, v_l$ and l tokens $t_1, t_2, \ldots, t_l$ such that for all $i \neq j$ either v_i is missing t_j or v_j is missing t_i. For a pair of nodes u and v, by union bound, the probability that u is missing t_v or v is missing t_u is at most $1/4 + 1/4 = 1/2$. Thus, the probability of E_l can be bounded as follows.

$$\Pr\left[E_l\right] \leq \binom{n}{l} \cdot \frac{k!}{(k-l)!} \cdot \left(\frac{1}{2}\right)^{\binom{l}{2}} \leq n^l \cdot k^l \frac{1}{2^{l(l-1)/2}} \leq \frac{2^{2l \log n}}{2^{l(l-1)/2}}.$$

In the above inequality, $\binom{n}{l}$ is the number of ways of choosing the l nodes that form the half-empty configuration, $k!/(k-l)!$ is the number of ways of assigning l distinct tokens, and $(1/2)^{\binom{l}{2}}$ is the upper bound on the probability for each pair $i \neq j$ that either v_i is missing t_j or v_j is missing t_i. For $l = 5 \log n$, $\Pr\left[E_l\right] \leq 1/n^2$. Thus, the largest half-empty configuration at the start of the first round, and hence at the start of any round, is of size at most $5 \log n$ with probability at least $1 - 1/n^2$. By Lemma 16, we thus obtain that the number of useful token exchanges in each round is at most $10 \log n$, with probability at least $1 - 1/n^2$.

Let M_i be the number of tokens that node i is missing in the initial distribution. Then M_i is a binomial random variable with $\mathbb{E}\left[M_i\right] = k/4$. By a straightforward Chernoff bound, we have the probability that node i misses less than $k/8$ tokens is

$$\Pr\left[M_i \leq \frac{k}{8}\right] = \Pr\left[M_i \leq \left(1 - \frac{1}{2}\right) \cdot \mathbb{E}\left[M_i\right]\right] \leq e^{-\frac{\mathbb{E}[M_i]\left(\frac{1}{2}\right)^2}{2}} = e^{-\frac{k}{32}}.$$

Therefore, the total number of tokens missing in the initial distribution is at least $n \cdot k/8 = \Omega(kn)$ with probability at least $1 - n/e^{\frac{k}{32}} \geq 1 - 1/n^2$ ($k \geq 100 \log n$). Since the number of useful tokens exchanged in each round is at most $10 \log n$, the number of rounds needed to complete k-gossip is $\Omega(kn/\log n)$ with high probability. □

Theorem 18 does not apply to certain natural initial distributions, such as one in which each token resides at exactly one node. While this class of token distributions has far fewer tokens distributed initially, the argument of Theorem 18 does not rule out the possibility that an algorithm, when starting from a distribution in this class, avoids the problematic configurations that arise in the proof. In the following, Theorem 20 extends the lower bound to this class of distributions.

Lemma 19. *From any distribution in which each token starts at exactly one node and no node has more than one token, any online token-forwarding algorithm for k-gossip needs $\Omega(kn/\log n)$ rounds against a strong adversary.*

Proof. We consider an initial distribution C where each token is at exactly one node, and no node has more than one token. Let C^* be an initial token distribution from which any online algorithm needs $\Omega(kn/\log n)$ rounds. The existence of C^* follows from Theorem 18. We construct a bipartite graph on two copies of V, V_1 and V_2. A node $v \in V_1$ is connected to a node $u \in V_2$ if in C^* u has all the tokens that v has in C. We will show below that this bipartite graph has a perfect matching with positive probability.

Given a perfect matching M, we can complete the proof as follows. For $v \in V_2$, let $M(v)$ denote the node in V_1 that got matched to v. If there is an algorithm A that runs in T rounds from starting state C, then we can construct an algorithm A^* that runs in the same number of rounds from starting state C^* as follows. First every node v deletes all its tokens except for those which $M(v)$ has in C. Then algorithm A^* runs exactly as A. Thus, the lower bound of Theorem 18, which applies to A^*, also applies to A.

It remains to prove that the above bipartite graph has a perfect matching. This follows from an application of Hall's Theorem. Consider a set of m nodes in V_2. We want to show their neighborhood in the bipartite graph is of size at least m. We show this condition holds by the following 2 cases. If $m < 3n/5$, let X_i denote the neighborhood size of node i. We know $\mathbb{E}[X_i] \geq 3n/4$. Then by Chernoff bound

$$\Pr[X_i < m] \leq \Pr[X_i < 3n/5] \leq e^{-\frac{(1/5)^2 \mathbb{E}[X_i]}{2}} = e^{-\frac{3n}{200}}$$

By union bound with probability at least $1 - n \cdot e^{-3n/200}$ the neighborhood size of every node is at least m. Therefore, the condition holds in the first case. If $m \geq 3n/5$, we argue the neighborhood size of any set of m nodes is V_1 with high probability. Consider a set of m nodes, the probability that a given token t is missing in all these m nodes is $(1/4)^m$. Thus the probability that any token is missing in all these nodes is at most $n(1/4)^m \leq n(1/4)^{3n/5}$. There are at most 2^n such sets. By union bound, with probability at least $1 - 2^n \cdot n(1/4)^{3n/5} = 1 - n/2^{n/5}$, the condition holds in the second case. □

Theorem 20. *From any distribution in which each token starts at exactly one node, any online token-forwarding algorithm for k-gossip needs $\Omega(kn/\log n)$ rounds against a strong adversary.*

Proof. In this theorem, we extend our proof in Lemma 19 to the inital distibution C where each token starts at exactly one node, but nodes may have multiple tokens. We prove this theorem by the following two cases.

First case, when at least $n/2$ nodes start with some token. This implies that $k \geq n/2$. Focus on the $n/2$ nodes with tokens. Each of them has at least one unique token. By the same argument used in Lemma 19, disseminating these $n/2$ distinct tokens to n nodes takes $\Omega(n^2/\log n)$ rounds. Thus, in this case the number of rounds needed is $\Omega(kn/\log n)$.

Second case, when less than $n/2$ nodes start with some token. In this case, the adversary can group these nodes together, and treat them as one super node. There is only one edge connecting this super node to the rest of the nodes. Thus, the number of useful token exchange provided by this super node is at most one in each round. If there exsits an algorithm that can disseminate k tokens in $o(kn/\log n)$ rounds, then the contribution by the super node is $o(kn/\log n)$. And by the same argument used in Lemma 19 we know dissemination k tokens to $n/2$ nodes (those start with no tokens) takes $\Omega(kn/\log n)$ rounds. Thus, the theorem also holds in this case. □

3.4 Subquadratic time offline token-forwarding algorithms

In this section, we give two centralized algorithms for the k-gossip problem in the offline model. We present an $O(\min\{n\sqrt{k \log n}, nk\})$ round algorithm in Section 3.4.1. Then we present a bicriteria $(O(n^\epsilon), \log n)$-approximation algorithm in Section 3.4.2, which means if L is the number of rounds needed by an optimal algorithm where one token is broadcast by every node per round, then our approximation algorithm will complete in $O(n^\epsilon L)$ rounds and the number of tokens broadcast by any node is $O(\log n)$ in any given round. Both of these algorithms uses a directed capacitated leveled graph constructed from the sequence of communication graphs which we call the *evolution graph.*

Evolution graph: Let V be the set of nodes. Consider a dynamic network of l rounds numbered 1 through l and let G_i be the communication graph for round i. The evolution graph for this network is a directed capacitated graph G with $2l+1$ levels constructed as follows. We create $2l+1$ copies of V and call them $V_0, V_2, \ldots, V_{2l}$. V_i is the set of nodes at level i and for each node v in V, we call its copy in V_i as v_i. For $i = 1, \ldots, l$, level $2i-1$ corresponds to the beginning of round i and level $2i$ corresponds to the end of round i. Level 0 corresponds to the network at the start. Note that the end of a particular round and the start of the next round are represented by different levels. There are three kinds of edges in the graph. First, for every round i and every edge $(u, v) \in G_i$, we place two directed edges with unit capacity each, one from u_{2i-1} to v_{2i} and another from v_{2i-1} to u_{2i}. We call these edges *broadcast edges* as they will correspond to broadcasting of tokens; the unit capacity on each such edge will ensure that only one token can be sent from a node to a neighbor in one round. Second, for every node v in V and every round i, we place an edge with infinite capacity from $v_{2(i-1)}$ to v_{2i}. We call these edges *buffer edges* as they ensure tokens can be stored at a node from the end of one round to the end of the next. Finally, for every node $v \in V$ and every round i, we also place an edge with unit capacity from $v_{2(i-1)}$ to v_{2i-1}. We call these edges as *selection edges* as they correspond to every node selecting a token out of those it has to broadcast in round i; the unit capacity ensures that in a given round a node must send the same token to all its neighbors. Figure 3.2 illustrates our construction, and Lemma 21 explains its usefulness.

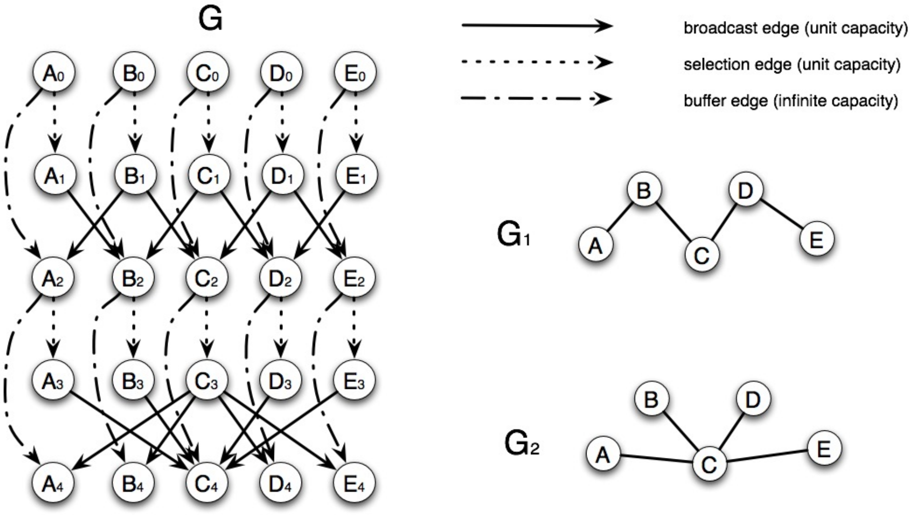


Figure 3.2: An example of how to construct the evolution graph from a sequence of communication graphs.

Lemma 21. *Let there be k tokens, each with a source node where it is present in the beginning and a set of destination nodes to whom we want to send it. It is feasible to send all the tokens to all of their destination nodes in a dynamic network using l rounds, where in each round a node can broadcast only one token to all its neighbors, if* *and only if k directed Steiner trees can be packed in the corresponding evolution graph with $2l+1$ levels respecting the edge capacities, one for each token with its root being the copy of the source node at level 0 and its terminals being the copies of the destination nodes at level $2l$.*

Proof. Assume that k tokens can be sent to all of their destinations in l rounds and fix one broadcast schedule that achieves this. We will construct k directed Steiner trees as required by the lemma based on how the tokens reach their destinations and then argue that they all can be packed in the evolution graph respecting the edge capacities. For a token i, we construct a Steiner tree T^i as follows. For each level $j \in \{0, \ldots, 2l\}$, we define a set S^i_j of nodes at level j inductively starting from level $2l$ backwards. S^i_{2l} is simply the copies of the destination nodes for token i at level $2l$. Once $S^i_{2(j+1)}$ is defined, we define S^i_{2j} (respectively S^i_{2j+1}) as: for each $v_{2(j+1)} \in S^i_{2(j+1)}$, include v_{2j} (respectively nothing) if token i has reached node v after round j, or include a node u_{2j} (respectively u_{2j+1}) such that u has token i at the end of round j which it broadcasts in round $j+1$ and (u, v) is an edge of G_{j+1}. Such a node u can always be found because whenever v_{2j} is included in S^i_{2j}, node v has token i by the end of round j which can be proved by backward induction staring from $j = l$. It is easy to see that S^i_0 simply consists of the copy of the source node of token i at level 0. T^i is constructed on the nodes in $\cup_{j=0}^{j=2l} S^i_j$. If for a vertex v, $v_{2(j+1)} \in S^i_{2(j+1)}$ and $v_{2j} \in S^i_{2j}$, we add the buffer edge $(v_{2j}, v_{2(j+1)})$ in T^i. Otherwise, if $v_{2(j+1)} \in S^i_{2(j+1)}$ but $v_{2j} \notin S^i_{2j}$, we add the selection edge (u_{2j}, u_{2j+1}) and broadcast edge $(u_{2j+1}, v_{2(j+1)})$ in T^i, where u was the node chosen as described above. It is straightforward to see that these edges form a directed Steiner tree for token i as required by the lemma which can be packed in the evolution graph. The argument is completed by noting that any unit capacity edge cannot be included in two different Steiner trees as we started with a broadcast schedule where each node broadcasts a single token to all its neighbors in one round, and thus all the k Steiner trees can be simultaneously packed in the evolution graph respecting the edge capacities.

Next assume that k Steiner trees as in the lemma can be packed in the evolution graph respecting the edge capacities. We construct a broadcast schedule for each token from its Steiner tree in the natural way: whenever the Steiner tree T_i corresponding to token i uses a broadcast edge (u_{2j-1}, v_{2j}) for some j, we let the node u broadcast token i in round j. We need to show that this is a feasible broadcast schedule. First we observe that two different Steiner trees cannot use two broadcast edges starting from the same node because every selection edge has unit capacity, thus there are no conflicts in the schedule and each node is asked to broadcast at most one token in each round.

Next we claim by induction that if node v_{2j} is in T^i, then node v has token i by the end of round j. For $j = 0$, it is trivial since only the copy of the source node for token i can be included in T^i from level 0. For $j > 0$, if v_{2j} is in T^i, we must reach there by following the buffer edge $(v_{2(j-1)}, v_{2j})$ or a broadcast edge (u_{2j-1}, v_{2j}). In the former case, by induction node v has token i after round $j-1$ itself. In the latter case, node u which had token i after round $j-1$ by induction was the neighbor of node v in G_j and u broadcast token i in round j, thus implying node v has token i after round j. From the above claim, we conclude that whenever a node is asked to broadcast a token in round j, it has the token by the end of round $j-1$. Thus the schedule we constructed is a feasible broadcast schedule. Since the copies of all the destination nodes of a token at level $2l$ are the terminals of its Steiner tree, we conclude all the tokens reach all of their destination nodes after round l. □

3.4.1 An $O(\min\{n\sqrt{k \log n}, nk\})$ round algorithm

Our algorithm is given in Algorithm 1 and analyzed in Lemma 22 and 23.

Lemma 22. *Let there be $k \leq n$ tokens at given source nodes and let v be an arbitrary node. Then, all the tokens can be sent to v using broadcasts in $O(n)$ rounds.*

Proof. By lemma 21, we will be done in $n + k$ rounds if we can show that k paths, one from every source vertex at level 0 to $v_{2(n+k)}$, can be packed in the corresponding evolution graph with $2(n + k) + 1$ levels respecting the edge capacities. For this, we consider the evolution graph and add to it a special vertex v_{-1} at level -1 and connect it to every source at level 0 by an edge of capacity 1. (Multiple edges get fused with corresponding increase in capacity if multiple tokens have the same source.) We claim that the value of the min-cut between v_{-1} and $v_{2(n+k)}$ is at least k. Before proving this, we complete the proof of the claim assuming this. By the max flow min cut theorem, the max flow between v_{-1} and $v_{2(n+k)}$ is at least k. Since we connected v_{-1} with each of the k token sources at level 0 by a unit capacity edge, it follows that unit flow can be routed from each of these sources at level 0 to $v_{2(n+k)}$ respecting the edge capacities. It is easy to see that this implies we can pack k paths, one from every source vertex at level 0 to $v_{2(n+k)}$, respecting the edge capacities.

To prove our claimed bound on the min cut, consider any cut of the evolution graph separating v_{-1} from $v_{2(n+k)}$ and let S be the set of the cut containing v_{-1}. If S includes no vertex from level 0, we are immediately done. Otherwise, observe that if $v_{2j} \in S$ for some $0 \leq j < (n + k)$ and $v_{2(j+1)} \notin S$, then the value of the cut is infinite as it cuts the

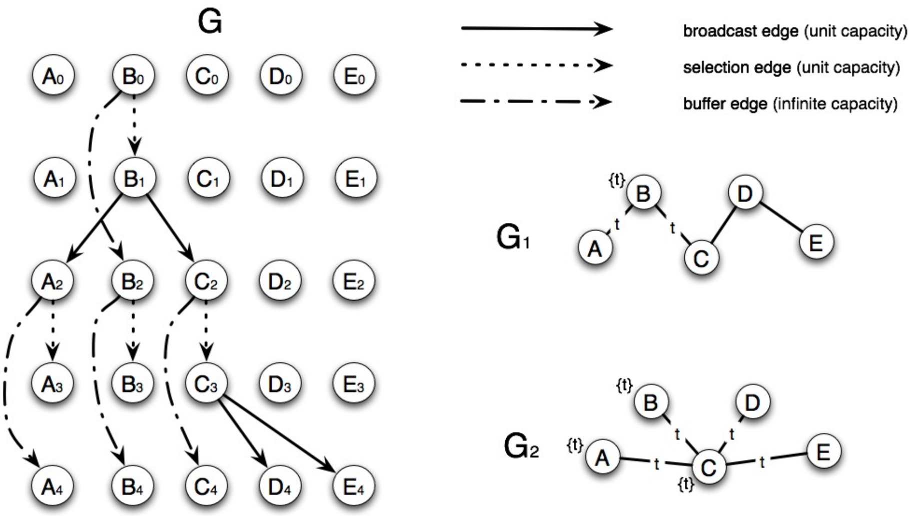

G
A0
B0
C0
D0
E0
A1
B1
C1
D1
E1
A2
B2
C2
D2
E2
A3
B3
C3
D3
E3
A4
B4
C4
D4
E4
broadcast edge (unit capacity)
selection edge (unit capacity)
buffer edge (infinite capacity)
G1
{t}
A
B
C
D
E
t
t
G2
{t}
{t}
{t}
t
t
t
t

Figure 3.3: An example of building directed Steiner tree in the evolution graph G based on token dissemination process. Token t starts from node B. Thus, the Steiner tree is rooted at B_0 in G. Since B_0 has token t, we include the infinite capacity buffer edge (B_0, B_2). In the first round, node B broadcasts token t, and hence we include the selection edge (B_0, B_1). Nodes A and C receive token t from B in the first round, so we include edges (B_1, A_2), (B_1, C_2). Now A_2, B_2, and C_2 all have token t. Therefore we include the edges (A_2, A_4), (B_2, B_4), and (C_2, C_4). In the second round, all of A, B, and C broadcast token t, we include edges (A_2, A_3), (B_2, B_3), (C_2, C_3). Nodes D and E receive token t from C. So we include edges (C_3, D_4) and (C_3, E_4). Notice that nodes A and B also receive token t from C, but they already have token t. Thus, we don't include edges (C_3, B_4) or (C_3, A_4).

buffer edge of infinite capacity out of v_{2j}. Thus we may assume that if $v_{2j} \in S$, then $v_{2(j+1)} \in S$. Also observe that since each of the communication graphs $G_1, \ldots, G_{n+k}$ are connected, if the number of vertices in S from level $2(j+1)$ is no more than the number of vertices from level $2j$ and not all vertices from level $2(j+1)$ are in S, we get at least a contribution of 1 in the value of the cut. But since the total number of nodes is n and $v_{2(n+k)} \notin S$, there must be at least k such levels, which proves the claim. □

Theorem 23. *Algorithm 1 solves the k-gossip problem using $O(\min\{n\sqrt{k \log n}, nk\})$ rounds with high probability in the offline model.*

Proof. It is trivial to see that if $k \leq \sqrt{\log n}$, then the algorithm will end in nk rounds and each node receives all the k tokens. Assume $k > \sqrt{\log n}$. By Lemma 22, all the

Algorithm 1 $O(\min\{n\sqrt{k\log n}, nk\})$ round algorithm in the offline model

Require: A sequence of communication graphs G_i, $i = 1, 2, \ldots$

Ensure: Schedule to disseminate k tokens.

```
if $k \leq \sqrt{\log n}$ then
    for each token $t$ do
        For the next $n$ rounds, let every node who has token $t$ broadcast the token.
    end for
else
    Choose a set $S$ of $2\sqrt{k\log n}$ random nodes.
    for each vertex in $v \in S$ do
        Send each of the $k$ tokens to vertex $v$ in $O(n)$ rounds.
    end for
    for each token $t$ do
        For the next $2n\sqrt{(\log n)/k}$ rounds, let every node who has token $t$ broadcast
        the token.
    end for
end if
```

tokens can be sent to all the nodes in S using $O(n\sqrt{k\log n})$ rounds. Now fix a node v and a token t. Since token t is broadcast for $2n\sqrt{(\log n)/k}$ rounds, there is a set S_v^t of at least $2n\sqrt{(\log n)/k}$ nodes from which v is reachable within those rounds. It is clear that if S intersects S_v^t, v will receive token t. Since the set S was picked uniformly at random, the probability that S does not intersect S_v^t is at most

$$\frac{\binom{n-2n\sqrt{(\log n)/k}}{2\sqrt{k\log n}}}{\binom{n}{2\sqrt{k\log n}}} < \left(\frac{n-2n\sqrt{(\log n)/k}}{n}\right)^{2\sqrt{k\log n}} \leq \frac{1}{n^4}.$$

Thus every node receives every token with probability $1 - 1/n^3$. It is also clear that the algorithm finishes in $O(n\sqrt{k\log n})$ rounds. □

Algorithm 1 can be derandomized using the standard technique of conditional expectations, shown in Algorithm 2. Given a sequence of communication graphs, if node u broadcasts token t for Δ rounds and every node that receives token t also broadcasts t during that period, then we say node v is within Δ *broadcast distance* to u if and only if v receives token t by the end of round Δ. Let S be a set of nodes, and $|S| \leq 2\sqrt{k\log n}$. We use $\Pr[u;S]\,T$ to denote the probability that the broadcast distance from node u to set X is greater than $2n\sqrt{(\log n)/k}$, where $X = S \cup \{\text{pick } 2\sqrt{k \log n} - |S| \text{ nodes uniformly at random from } V \setminus T\}$, and $P(S,T)$ denotes the sum, over all u in V, of $\Pr[u;S]\,T$.

Algorithm 2 Derandomized algorithm for Step 6 in Algorithm 1

Require: A sequence of communication graphs G_i, $i = 1, 2, \ldots$, and $k \geq \sqrt{\log n}$

Ensure: A set of $2\sqrt{k \log n}$ nodes S such that the broadcast distance from every node u to S is within $2n\sqrt{(\log n)/k}$.

```
Set $S$ and $T$ be $\emptyset$.
for each $v \in V$ do
    $T = T \cup \{v\}$
    if $P(S \cup \{v\}, T) \leq P(S,T)$ then
        $S = S \cup \{v\}$
    end if
end for
return $S$
```

Lemma 24. *The set S returned by Algorithm 2 contains at most $2\sqrt{k \log n}$ nodes, and the broadcast distance from every node to S is at most $2n\sqrt{(\log n)/k}$.*

Proof. Let us view the process of randomly selecting $2\sqrt{k \log n}$ nodes as a computation tree. This tree is a complete binary tree of height n. There are $n+1$ nodes on any root-leaf path. The level of a node is its distance from the root. The computation starts from the root. Each node at the ith level is labeled by $b_i \in \{0, 1\}$, where 0 means not including node i in the final set and 1 means including node i in the set. Thus, each root-leaf path, $b_1 b_2 \ldots b_n$, corresponds to a selection of nodes. For a node a in the tree, let S_a (resp., T_a) denote the sets of nodes that are included (resp., lie) in the path from root to a.

By Theorem 23, we know that for the root node r, we have $P(\emptyset, S_r) = P(\emptyset, \emptyset) \leq 1/n^3$. If c and d are the children of a, then $T_c = T_d$, and there exists a real $0 \leq p \leq 1$ such that for each u in V, $\Pr[u;S_a]\,T_a$ equals $p \Pr[u;S_c]\,T_c + (1-p)\Pr[u;S_d]\,T_d$. Therefore, $P(S_a, T_a)$ equals $pP(S_c, T_c) + (1-p)P(S_d, T_d)$. We thus obtain that $\min\{P(S_c, T_c), P(S_d, T_d)\} \leq P(S_a, T_a)$. Since we set S to be X in $\{S_c, S_d\}$ that minimizes $P(X, T_c)$, we maintain the invariant that $P(S,T) \leq 1/n^3$. In particular, when the algorithm reaches a leaf l, we know $P(S_l, V) \leq 1/n^3$. But a leaf l corresponds to a complete node selection, so that $\Pr[u;S_l]\,V$ is 0 or 1 for all u, and hence $P(S_l, V)$ is an integer. We thus have

$P(S_l, V) = 0$, implying that the broadcast distance from node u to set S_l is at most $2n\sqrt{(\log n)/k}$ for every l. Furthermore, $|S_l|$ is $2k\sqrt{\log n}$ by construction.

Finally, note that Step 4 of Algorithm 2 can be implemented in polynomial time, since for each u in V, $\Pr[u;S]\,T$ is simply the ratio of two binomial coefficients with a polynomial number of bits. Thus, Algorithm 2 is a polynomial time algorithm with the desired property. □

3.4.2 An $(O(n^\epsilon), \log n)$-approximation algorithm

Here we introduce an $(O(n^\epsilon), \log n)$-approximation algorithm for the k-gossip problem in the offline model. This means, if the k-gossip problem can be solved on any n-node dynamic network in L rounds, then our algorithm will solve the k-gossip problem on any dynamic network in $O(n^\epsilon L)$ rounds, assuming each node is allowed to broadcast $O(\log n)$ tokens, instead of one, in each round. Our algorithm is an LP based one, which makes use of the evolution graph defined earlier. The following is a straightforward corollary of Lemma 21.

Corollary 25. *The k-gossip problem can be solved in l rounds if k directed Steiner trees can be packed in the corresponding evolution graph, where for each token, the root of its Steiner tree is a source node at level 0, and the terminals are all the nodes at level $2l$.*

Packing Steiner trees in general directed graphs is NP-hard to approximate even within $\Omega(m^{1/3-\epsilon})$ for any $\epsilon > 0$ [48], where m is the number of edges in the graph. Thus, our algorithm focuses on solving Steiner tree packing problem with relaxation on edge capacities, allowing the capacity to blow up by a factor of $O(\log n)$. First, we write down the LP for the Steiner tree packing problem (maximizing the number of Steiner trees packed with respect to edge capacities). Let $\mathcal{T}$ be the set of all possible Steiner trees, and c_e be the capacity of edge e. For each Steiner tree $T \in \mathcal{T}$, we associate a variable x_T with it. If $x_T = 1$, then Steiner tree T is in the optimal solution; if $x_T = 0$, it's not. After relaxing the integral constraints on x_T's, we have the following LP, referred to as $\mathcal{P}$ henceforth. Let $F(\mathcal{P})$ denote the optimal fractional solution for $\mathcal{P}$.

$$\begin{array}{rrl} \max & \sum_{T\in\mathcal{T}} x_T & \\ \text{s.t.} & \sum_{T:e\in T} x_T \le c_e & \forall e \in E \\ & x_T \ge 0 & \forall T \in \mathcal{T} \end{array}$$

Lemma 26 ([48])**.** *There is an $O(n^\epsilon)$-approximation algorithm for the fractional maximum Steiner tree packing problem in directed graphs.*

Let L be the number of rounds that an optimal algorithm uses with every node broadcasting at most one token per round. We give an algorithm that takes $O(n^\epsilon L)$ rounds with every node broadcasting $O(\log n)$ tokens per round. Thus ours is an $(O(n^\epsilon), O(\log n))$ bicriteria approximation algorithm, shown in Algorithm 3.

Algorithm 3 $(O(n^\epsilon), O(\log n))$-approximation algorithm

```
Require: A sequence of communication graphs $G_1, G_2, \ldots$
Ensure: Schedule to disseminate $k$ tokens.
Initialize the set of Steiner trees $\mathcal{S} = \emptyset$.
for $i = 1 \to 2n^\epsilon$ do
    Find $L^*$ such that with the evolution graph $G$ constructed from level 0 to level $2L^*$, the approximate value for $F(\mathcal{P})$ is $k/n^\epsilon$. In this step, we use the algorithm of [48] to approximate $F(\mathcal{P})$.
    Let $x_T^*$ be the value of the variable $x_T$ in the solution from step 3. The number of non-zero $x_T^*$'s is polynomial with respect to $k$. Using randomized rounding, with probability $x_T^*$ include $T$ in the solution, $\mathcal{S} = \mathcal{S} \cup \{T\}$. Otherwise, don't include $T$.
    Remove communication graphs $G_1, G_2, \ldots, G_{L^*}$ from the sequence, and reduce the remaining graphs' indices by $L^*$.
end for
Use Corollary 25 to convert the set of Steiner trees $\mathcal{S}$ into a token dissemination schedule.
```

Theorem 27. *Algorithm 3 achieves an $O(n^\epsilon)$ approximation to the k-gossip problem while broadcasting $O(\log n)$ tokens per round per node, with high probability.*

Proof. We show the following three claims: (i) In Step 7, $|\mathcal{S}| \geq k$ with probability at least $1 - 1/e^{k/4}$. This is the correctness of Algorithm 3, saying it can find the schedule to disseminate all k tokens. (ii) The number of rounds in the schedule produced by Algorithm 3 is at most $O(n^\epsilon)$ times the optimal one. (iii) In the token dissemination schedule, the number of tokens sent over an edge is $O(\log n)$ in any round with high probability.

First, we prove claim (i). Let X_i denote the sum of non-zero x_T^*'s in iteration i. $X = \sum_{i=1}^{2n^\epsilon} X_i$. We know $\mathbb{E}[X_i] = k/n^\epsilon$. Thus, $\mathbb{E}[X] = 2n^\epsilon k/n^\epsilon = 2k$, which is the expected number of Steiner trees in set $\mathcal{S}$. By Chernoff bound, we have

$$\Pr\left[X \le k\right] = \Pr\left[X \le \left(1 - \frac{1}{2}\right)\mathbb{E}\left[X\right]\right] \le e^{-\frac{(1/2)^2 \mathbb{E}[X]}{2}} = e^{-\frac{(1/2)^2 \cdot 2k}{2}} = \frac{1}{e^{k/4}}$$

Thus, $|\mathcal{S}| \ge k$ with probability at least $1 - 1/e^{k/4}$ in Step 7.

Next we prove claim (ii). Let L denote the number of rounds needed by an optimal algorithm. Since in Step 3 we used the $O(n^\epsilon)$-approximation algorithm in [48] to solve $F(\mathcal{P})$, we know $L^* \le L$. There are $2n^\epsilon$ iterations. Thus, the number of rounds needed by Algorithm 3 is at most $2n^\epsilon L^* \le 2n^\epsilon L$, which is an $O(n^\epsilon)$-approximation on the number of rounds.

Lastly we prove claim (iii). When Algorithm 3 does randomized rounding in Step 4, some constraint $\sum_{T:e\in T} x_T \le c_e$ in $\mathcal{P}$ may be violated. In the evolution graph, $c_e = 1$. Let Y denote the sum of x_T^*'s in this constraint. We have $\mathbb{E}\left[Y\right] \le c_e = 1$. By Chernoff bound,

$$\begin{aligned} \Pr\left[Y \ge \mathbb{E}\left[Y\right] + \log n\right] &= \Pr\left[Y \ge \left(1 + \frac{\log n}{\mathbb{E}\left[Y\right]}\right)\mathbb{E}\left[Y\right]\right] \\ &\le e^{-\mathbb{E}[Y]\left[\left(1+\frac{\log n}{\mathbb{E}[Y]}\right)\ln\left(1+\frac{\log n}{\mathbb{E}[Y]}\right) - \frac{\log n}{\mathbb{E}[Y]}\right]} \le \frac{1}{n^{\log\log n}} \end{aligned}$$

Thus, the number of tokens sent over a given edge is $O(\log n)$ with probability at least $1 - 1/n^{\log\log n}$. Since there are only polynomial number of edges, no edge will carry more than $O(\log n)$ tokens in a single round with high probability. □

3.5 Conclusion and open questions

In this paper, we studied the power of token-forwarding algorithms for gossip in dynamic networks. We showed a lower bound of $\Omega(nk/\log n)$ rounds for any online token forwarding algorithm against a strong adversary; our bound matches the known upper bound of $O(nk)$ up to a logarithmic factor. We note that our lower bound also extends to randomized algorithms if the adversary is allowed to be adaptive; that is, the adversary is allowed to make its decision in each step with knowledge of the random coin tosses made by the algorithm in that step (but without knowledge of the randomness used in future steps). This leaves us with an important open question: what is the complexity of randomized online token-forwarding algorithms against a weak adversary that is unaware of the randomness used by the algorithm in each round? Furthermore, for small token sizes (e.g., $O(\log n)$ bits) even the best (randomized) online algorithm we know based on network coding takes $O(nk/\log n)$ rounds [73]. In contrast, we show that in the offline setting there exist centralized token-forwarding algorithms that run in $O(n^{1.5}\sqrt{\log n})$ time. An interesting open problem is to obtain tight bounds on offline token-forwarding algorithms.

Chapter 4

Controlling negative diffusion

In Chapter 2 and Chapter 3, we have studied how to enable positive diffusion under both organic and adversarial dynamics. In this chapter, we switch gear to controlling negative diffusion.

Over the recent decades, there has been an explosive growth in the use of personal digital devices of various kinds. This has, unfortunately, been accompanied by significant increase in worm attacks. While, effective anti-virus software and patches are readily available, the average user is very independent and does not often loading these latest patches due to software cost and other efforts involved. Also if enough other nodes in the network are secured, the likelihood of a specific device getting infected would go down, leading to a natural game theoretic scenario. Aspnes et al [15] introduced an innovative game for modeling the containment of the spread of viruses and worms (security breaches) in a network. In this model, nodes choose to install anti-virus software or not on an individual basis while the viruses or worms start from a node chosen uniformly at random and spread along paths consisting of insecure nodes. They showed the surprising result that a pure Nash Equilibrium always exists when all nodes have identical installation costs and identical infection costs.

In this chapter we present a substantial generalization of the model of [15] that allows for arbitrary security and infection costs, and arbitrary distributions for the starting point of the attack. More significantly, our model GNS(d) incorporates a network locality parameter d which represents a hop-limit on the spread of infection as accounted for in the strategic decisions, due to either the intrinsic nature of the infection or the extent of neighborhood information that is available to a node.

We determine that the network locality parameter plays a key role in the existence of pure Nash equilibria (NE): local ($d = 1$) and global games ($d = \infty$) have pure NE, while for GNS(d) games with $1 < d < \infty$, pure NE may not exist, and in fact, it is NP-complete to determine whether a given instance has a pure NE. For local and global games, we also characterize the price of anarchy in terms of the maximum degree and vertex expansion of the contact network; these suggest natural heuristics to aid a network planner in enforcing efficient equilibria.

We design a general LP-based framework for approximating the NP-complete problem of finding a socially optimal configuration in our game. Our framework yields a $2d$-approximation for general GNS(d) games, and an $O(\log n)$-approximation for the global model where n is the number of network nodes; the latter result improves on the approximation bound of $O(\log^{1.5} n)$ of [15] achieved for a special case of our global model.

We study the characteristics of NE and the quality of our approximations empirically in two distinct classes of graphs: random geometric graphs and power law graphs. We find that in local and global games on these real-world networks, best response dynamics converge in linear or sub-linear time and have costs comparable to the social optimum. Finally, we study the performance of our approximation algorithms, and find that the approximation guarantees with respect to social cost are much better in practice than our theoretical bounds.

4.1 Related Work

Non-cooperative game theory has been used in analyzing a number of problems in traffic and communication networks, e.g., routing [117], topology control and network formation [59, 107] and security [71, 112]. The basic questions of interest have usually been about the existence and the structure of Nash equilibria and the price of anarchy, which is the worst case cost of a Nash equilibrium to the social optimum, as defined formally later. See [111] for a good introduction on the use of game theoretic techniques for networking applications.

Several formulations have been proposed for analyzing network security problems and the spread of epidemics in networks [15, 16, 35, 67, 71, 94, 126]. This thesis directly builds on the formulation of Aspnes et al. [15], who model the risk of infection for an insecure node v as the probability that the initial infection, which is assumed to originate at a node chosen uniformly at random, starts in the same component as v in the subgraph induced by v and the other insecure nodes. They show the surprising result that pure Nash equilibria always exist in such games. They also establish a high price of anarchy and give an $O(\log^{1.5} n)$ approximation algorithm for computing the social optimum, where n is the number of nodes in the network. Their approximation algorithm uses an $O(\sqrt{\log n})$-approximation for the sparsest cut problem [14], which is based on a semidefinite programming relaxation of the problem. In this thesis, we are able to give a much simpler LP-based approximation algorithm using the vertex multicut problem, which improves the approximation ratio to $O(\log n)$ and also applies to a more general model. Another direction of work is based on SIS models for the worm spread, e.g., the n-intertwined model [112]. In this model, nodes are in two states - susceptible or infected. Each infected node spreads the infection to its neighbors with some probability. Another closely related class of models is that of Interdependent Security games (IDS) [81], which is similar to our model for the special case of $d = 1$. One crucial technical difference between the two models, which leads to two different games, is the assumption about the initial infection: in IDS, it is assumed to originate independently at different nodes, while in our GNS(1) model, we assume an initial location is selected according to a given probability distribution.

Our formulation of generalized network security games is largely motivated by mechanisms to protect communication networks. Some of our model and results, especially the lower bound results, however, also apply equally well to the spread of diseases and the protection of communities through vaccinations. The pure Nash equilibria correspond to stable points in the space of vaccination decisions made by individuals, and our approximation algorithms yield public policies for vaccination that well-approximate the social welfare. There is considerable work in epidemiology, both from a game-theoretic perspective, as well as on the analysis of disease spreads through SIR and SIS models [32, 31, 85, 30, 33]. The game-theoretic models adopted in these studies, however, do not consider the impact of the underlying contact network. Furthermore, there is little work on quantifying the effect of locality (in disease spread or in information availability).

4.2 Model and Definitions

In this section, we present our game-theoretic model for network security.

Contact Graph. Let V denote the set of users/devices (henceforth, referred to as *nodes*), each of which is assumed to be an autonomous player. Let G denote the underlying contact graph over the node set V; an edge $(u, v) \in G$ indicates that nodes u and v are directly connected, so that if node u is infected by a worm it can potentially spread to node v. Let $N(v)$ denote the set of neighbors of v in G. We will frequently work with certain subgraphs of G, for which we introduce the following notation. For any undirected graph H and subset S of vertices of H, we let $H[S]$ denote the subgraph of H induced by the vertices in S.

Strategies. The strategy for each node v is the decision of whether to install an anti-virus software or not; we use a variable $a_v \in [0, 1]$ to denote the probability of securing the device. In this paper, we focus on *pure* strategies, i.e., $a_v \in \{0, 1\}$. Let $\vec{a}$ denote the strategy vector of all nodes. Following [15], the *attack graph*, $G_{\vec{a}}$, is the subgraph of the contact graph induced by the set of insecure nodes according to $\vec{a}$. For notational convenience, let $\vec{a}[v/x]$ be the strategy vector obtained by replacing a_v by x in the vector $\vec{a}$.

Infection model. We assume that the infection is initiated at a node chosen from V according to an arbitrary probability distribution. Let w_v denote the probability that node v is chosen as the initial infection point; for convenience, we introduce the notation $w(S)$ to denote the sum of w_v over all v in S. We parameterize the infection model by d, the maximum number of hops over which the probability of infection spread is taken into account in the decision making. Thus, for a given contact graph G and strategy vector $\vec{a}$, an infection originating at node v infects node u if and only if u is within d hops of v in $G_{\vec{a}}$. Since G is fixed and d is clear from the context, denote by $S_v(\vec{a})$ the set of nodes that are within d hops of v in $G_{\vec{a}[v/0]}$. For a given strategy vector $\vec{a}$, therefore, the probability that node v gets attacked in this model (denoted by $p_v(\vec{a})$) is $w(S_v(\vec{a}))$.

Generalized Network Security Game GNS(d). We now present our model for a generalized network security game GNS(d), parameterized by the hop-limit d in the infection model. The game GNS(d) is specified by a contact graph G, initial infection probability distribution w, and two costs per network node. Let C_v denote the security cost (installing an anti-virus software) of user v; we assume the software is fool-proof so that secure nodes do not get attacked. Let L_v denote the infection cost of user v (recovering from a worm attack in case an insecure node v gets attacked). Then, the cost to node v is defined as

$$\text{cost}_v\left(\bar{a}\right) = a_v C_v + (1 - a_v) L_v \cdot p_v\left(\bar{a}\right).$$

A pure Nash equilibrium (henceforth, pure NE) is a strategy vector $\vec{a}$ such that no node v has any incentive to switch his strategy, if all other nodes' strategies are fixed. $\vec{a}$ is a Nash equilibrium if $\text{cost}_v(\vec{a}[v/x]) \geq \text{cost}_v(\vec{a})$ for $x \in \{0, 1\}$. Therefore, a pure NE is a natural configuration to aim for in a non-cooperative game. It is easy to verify that the following characterization of a pure NE (shown in [15] for the special case where G is the complete graph) holds.

Lemma 28. *For $v \in V$, let $t_v = C_v/L_v$. A strategy vector $\vec{a} \in \{0, 1\}^n$ is a pure NE if the following conditions hold: (i) for all v such that $a_v = 0$, $w(S_v(\vec{a})) \leq t_v$, and (ii) for all v such that $a_v = 1$, $w(S_v(\vec{a}[v/0])) > t_v$.*

Social cost. The total social cost of a strategy profile is the sum of the individual costs, which is $\text{cost}\left(\bar{a}\right) = \sum_{v=1}^{n} \text{cost}_v\left(\bar{a}\right)$. A socially optimum strategy is a vector $\vec{a}$ that minimizes this cost - this is not necessarily (and is not usually) a pure NE. Therefore, the cost of a pure NE relative to the social cost is an important measure; the maximum such ratio (i.e., over all possible pure NE) is also known as the *price of anarchy* [88].

For convenience, Table 4.1 summarizes our notations.

4.3 Nash equilibria

4.3.1 The local infection model: $d = 1$

For the local infection model, we show that a pure NE always exists. Our proof is by a reduction to a result of Borodin et al. [41] on existence of subgraphs with restricted degree sequences; their result is based on a potential function argument.

Theorem 29. *Every* GNS(1) *instance has a pure NE.*

Proof. We first define two functions $a : V \to \mathbb{R}$ and $b : V \to \mathbb{R}$. For each $v \in V$, $a(v) = w(N(v)) - \frac{C_v}{L_v} + w(v)$ and $b(v) = \frac{C_v}{L_v} - w(v)$. We argue next, using a generalization

Table 4.1: A list of notations.

Notations	Explanation
G	Contact graph.
$G[S]$	Subgraph of G induced by the vertices in S.
C_v	Security cost for node v
L_v	Infection cost for node v
$\vec{a}$	Strategy vector of nodes.
$G_{\vec{a}}$	Attack graph, i.e. the subgraph of the contact graph induced by the set of insecure nodes according to $\vec{a}$.
$\vec{a}[v/x]$	Strategy vector obtained by replacing a_v by x in the vector $\vec{a}$.
$S_v(\vec{a})$	Set of nodes that are within d hops of v in $G_{\vec{a}[v/0]}$.
w_v	Probability that node v is chosen as the initial infection point.
$w(S)$	Sum of w_v over all v in S.
$\text{cost}_v(\vec{a})$	Cost to node v given strategy vector $\vec{a}$.
$\text{GNS}(d)$	Generalized network security game parameterized by the disease hop limit d.

of an argument due to [41], that there exists a partition $V = A \cup B$ such that for each $v \in A$, we have $w(A \cap N(v)) \leq a(v)$ and for each $v \in B$, we have $w(B \cap N(v)) \leq b(v)$. Consider the following function that defines a potential for each partition (A, B).

$$\begin{aligned} R(A,B) = \quad & \textstyle\sum_{v\in A} w(v)\,(w(A \cap N(v)) - 2a(v)) \\ & + \textstyle\sum_{v \in B} w(v)\,(w(B \cap N(v)) - 2b(v)) \end{aligned}$$

Among all the partitions, we take a partition (A^*, B^*) minimizing R and assert that (A^*, B^*) is the partition we need. Suppose that a vertex x belongs to A^*, and $w(A^* \cap N(x)) > a(x)$. Now we move x from A^* to B^* to obtain the partition $(A' = A^* \setminus \{x\}, B' = B^* \cup \{x\})$. Because $a(x) + b(x) \geq w(N(x))$, we have $w(N(x) \cap B^*) \leq b(x)$. It is easy to verify that $R(A^*, B^*) - R(A', B')$ equals $w(x)\,(w(N(x) \cap A^*) - 2a(x)) + w(x)w(N(x) \cap A^*) - w(x)\,(w(N(x) \cap B^*) - 2b(x)) - w(x)w(N(x) \cap B^*) = 2w(x)\,(w(N(x) \cap A^*) - a(x)) - 2w(x)\,(w(N(x) \cap B^*) - b(x)) > 0$. This means $R(A^*, B^*) > R(A', B')$, which is a contradiction. A similar inequality follows if there is a vertex $x \in B^*$ with $w(B^* \cap N(x)) > b(x)$. Therefore, such a vertex x doesn't exist implying that (A^*, B^*) is the desired partition.

Given such a partition (A, B), we establish the existence of pure NE. Let $\vec{a}$ be a strategy vector with $a_v = 1$ for all $v \in A$ and $a_v = 0$ for all $v \in B$; i.e., A denotes the set of secure nodes. Then, we argue that $\vec{a}$ is indeed a pure NE. First consider the case where $v \in A$. Then v is secure and pays cost C_v. If v changes strategy, its expected infection cost is $L_v\,(w(N(v) \cap B) + w(v))$. Since $v \in A$, we have $w(N(v) \cap A) \leq a(v) = w(N(v)) - C_v/L_v + w(v)$. Therefore, $C_v \leq L_v\,(w(N(v) \cap B) + w(v))$, i.e. v won't change its strategy. Next consider $v \in B$. Then v is not secure and its expected infection cost is $L_v\,(w(N(v) \cap B) + w(v))$. If v changes strategy, its cost is C_v. Since $v \in B$, we have $w(N(v) \cap B) \leq b(v) = C_v/L_v - w(v)$. Therefore, $L_v\,(w(N(v) \cap B) + w(v)) \leq C_v$, i.e. v won't change its strategy. Thus it follows that $\vec{a}$ is a Nash equilibrium.

□

When the security and infection costs are uniform, we show that for the case of $d = 1$, the maximum ratio of the cost of a pure NE to the social optimum is bounded by the maximum degree.

Lemma 30. *When security and infection costs are uniform, and $w_v = 1/n$ $\forall v$, the price of anarchy in* GNS*(1) is at most $\Delta + 1$, where Δ is the maximum degree of the contact graph.*

Proof. Let C and L denote the security and infection costs, respectively. Suppose $C > L(\Delta + 1)/n$. Then no node is secured in any pure NE and therefore, the cost of any pure NE is at most $L(\Delta + 1)$. In the optimum strategy, each node has a cost of C if it is secured, or at least L/n otherwise. Therefore the optimal cost is at least L, and the lemma follows in this case.

Next, consider the case $C \leq L(\Delta + 1)/n$. In any pure NE, any node has cost at most C, and therefore the cost of a pure NE is at most Cn. If $C \leq L/n$, the optimum cost is also Cn, and therefore, we assume $C \geq L/n$. In an optimum solution, each node has cost at least L/n, and therefore, the optimal cost is at least L. Therefore, the price of anarchy in this case is at most $\Delta + 1$. □

4.3.2 The global infection model: $d = \infty$

In this section, we consider the global model ($d = \infty$); thus, any node v is capable of infecting any other node u as long there is a path of insecure nodes between v and u in the contact graph G. In this special case, our model is a generalization of the model of [15] in that we allow different security costs, infection costs, and initial infection probabilities.

Theorem 31. *Every* GNS(∞) *instance has a pure NE.*

Proof. Let $t_v = C_v/L_v$; we refer to t_v as the threshold for v. We relabel the n nodes so that $t_1 \geq t_2 \geq \cdots \geq t_n$, where we break ties arbitrarily. Given a strategy vector $\vec{a}$, we say that a secure node v is *happy* if $w(S_v(\vec{a}[v/0])) > t_v$, and *unhappy* otherwise. Similarly, an insecure node v is *happy* if $w(S_v(\vec{a})) \leq t_v$, and *unhappy* otherwise. Recall that when $d = \infty$, $S_v(\vec{a})$ is the set of nodes that can reach v in $G_{\vec{a}}$.

Consider the following potential function.

$$\hat{\Phi}(\vec{a}) = (\Phi_1(\vec{a}), \Phi_2(\vec{a}), \ldots, \Phi_n(\vec{a}))$$

where $\Phi_v(\vec{a})$ is 0 if v is secure, -1 if v is insecure and happy, and 1 otherwise. We next show this potential always lexicographically decreases. There are two cases:

1. Some node v switches from being an insecure unhappy node to being a secure happy node, changing the strategy vector from $\vec{a}$ to $\vec{b}$. In this case $w(S_v(\vec{a})) > t_v$. Since the set of secure nodes in $\vec{b}$ is a superset of the set of secure nodes in $\vec{a}$, it follows that for any node u, $w(S_u(\vec{b})) \leq w(S_u(\vec{a}))$; it thus follows that no insecure happy node in $\vec{a}$ can become unhappy in $\vec{b}$. Therefore, the vth component of the potential decreases by 1, while none of the other components increases.

2. Some node v switches from being secure to not being secure, changing the strategy vector from $\vec{a}$ to $\vec{b}$. In this case, $w(S_v(\vec{b})) \leq t_v$. We thus have the vth component of

the potential changing from 0 to -1. Consider any node $u \neq v$. If u is secure, then the uth component of the potential is unchanged. Otherwise, consider two cases. If v and u are in different connected components, then $w(S_u(\vec{b})) = w(S_u(\vec{a}))$, implying that the uth component of the potential is unchanged. If v and u are in the same connected component, then $w(S_u(\vec{b})) = w(S_v(\vec{b}))$; thus, if u is happy in $\vec{a}$ but unhappy in $\vec{b}$, then it must be the case that $t_u < t_v$, implying that $u > v$. Thus, the only components of the potential that can increase are the components greater than v, implying that the potential decreases lexicographically.

Since the value of each column in the potential vector is between -1 and 1, and this potential vector lexicographically decreases, we conclude that this process converges to a pure Nash equilibrium (in fact, in at most 3^n steps). □

Even when the security and infection costs are uniform, [15] showed that the price of anarchy is $\Omega(n)$. We give a more precise characterization in terms of the vertex expansion of the contact graph. For any graph H over vertex set V, the vertex expansion $\alpha(H)$ is defined as the largest number c such that for any subset V' of the vertices such that $|V'| \leq |V|/2$, the set of vertices in $V \setminus V'$ that are adjacent to a vertex in V' is at least $c|V'|$.

Lemma 32. *When security and infection costs are uniform, and $w_v = 1/n$ $\forall v$, the price of anarchy in any* GNS(∞) *game is $O(1/\alpha(G))$.*

Proof. First we calculate the lower bound for social optimum. Let $\vec{a}$ be the strategy vector of a social optimum, and $S_1, S_2, \ldots, S_m$ denote the connected components in $G_{\vec{a}}$. Without loss of generality, we can assume $|S_1| \leq |S_2| \leq \cdots \leq |S_m|$. We consider the following 3 cases:

1. $\sum_i |S_i| < n/2$, where n is the total number of nodes in G. In this case more than half of the nodes are secure. Thus, social optimal cost is at least $Cn/2$.

2. $\sum_i |S_i| \geq n/2$ and $|S_m| \geq n/4$. Then social optimal cost is at least $\sum_{v \in S_m} \mathrm{cost}_v(\vec{a}) \geq \frac{n}{4} L \frac{n/4}{n} = Ln/16$.

3. $\sum_i |S_i| \geq n/2$ and $|S_m| < n/4$. Then there must be a j such that $\sum_{i \leq j} |S_i| \geq n/4$. Let $S = \cup_{i \leq j} S_i$. Then the number of neighbors of set S in G is at least $\alpha(G)|S| \geq \alpha(G)n/4$. This implies social optimal cost is at least $C\alpha(G)n/4$.

Therefore, the lower bound for social optimum is $\min\{Cn/2, Ln/16, C\alpha(G)n/4\}$.

Next we calculate the upper bound for NE cost. Let $\vec{a}$ be the strategy vector of a NE. Again, let $S_1, S_2, \ldots, S_m$ denote the connected components in $G_{\vec{a}}$. $|S_1| \leq |S_2| \leq \cdots \leq |S_m|$. We consider the following 2 cases.

1. $L \leq C$. In this case no one is going to be secure in NE, which implies its cost is nL. The ratio between NE and the social optimum is no more than $\max\{2, 16, 4/\alpha(G)\}$.

2. $L > C$. The cost of NE is no more than $\sum_i L|S_i|^2/n + Cn$. Because this is a NE, for those who choose to be insecure, $L|S_i|/2 \leq C$. Therefore, we have $\sum_i L|S_i|^2/n + Cn \leq \sum_i C|S_i| + Cn \leq 2Cn$. The ratio between NE and the social optimum is no more than $\max\{4, 32, 8/\alpha(G)\}$.

Putting these 2 cases together completes the proof of this lemma. □

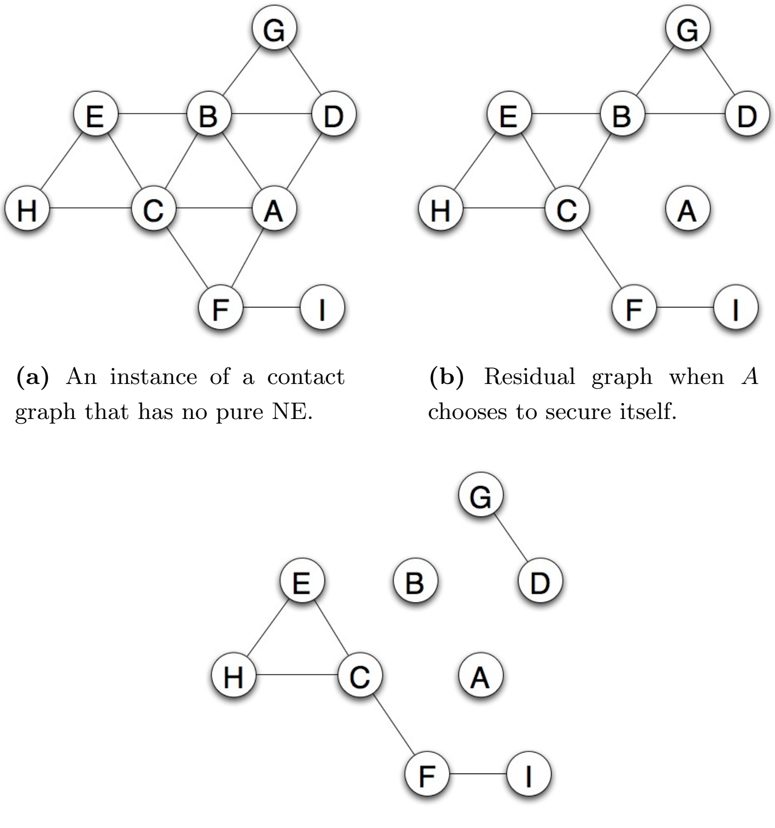
4.3.3 The d-neighborhood infection model: $d > 1$

Having established the existence of a pure NE for every instance of the generalized network security game in both the local and the global models, a natural question is whether pure NE exist for the entire spectrum of d in between these two extremes. In this section, we show that for any $1 < d < \infty$, there exist instances of GNS(d) for which there are no pure NE. Furthermore, it is NP-complete to determine whether a pure NE exists for a given instance. We first present the non-existence result which also provides the basis for the NP-hardness reduction.

Lemma 33. *For any fixed d, $1 < d < \infty$, there exists an instance of* GNS(d) *in which no pure NE exists.*

Proof. We first consider the case $d = 2$. Consider the instance defined by the contact graph in Figure 4.1a. $w_v = 1/n$ for all node v. We set the infection cost to be identical, say L, for all nodes. For nodes D through I, we set the security cost to be high enough so that in any equilibrium they are all insecure. That leaves nodes A, B, and C, for whom we set the security cost such that $9C_v/L = 7 + \epsilon$ for v in {A,B,C}; thus, in any pure NE $\vec{a}$, node v in {A, B, C} is secure if and only if $|S_v(\vec{a}[v/0])| \geq 7 + \epsilon$. We now consider four cases. If all of A, B, and C are insecure in $\vec{a}$, then we do not have a pure NE since $|S_v(\vec{a}[v/0])| = 9$ for each v in {A, B, C}. If exactly one of A, B, or C – say A – is secure, as shown in Figure 4.1b, then B won't change its strategy since $|S_B(\vec{a})| = 7$, but C will change its strategy since $|S_C(\vec{a})| = 8$ (Notice C can reach I, but B cannot). If exactly two of A, B, C – say A and B – are secure, as shown in Figure 4.1c, then

B will change its strategy since $|S_B(\vec{a}[B/0])| = 7$. Finally, if all three are secure, then none of A, B, or C will stick to its current strategy since $|S_v(\vec{a}[v/0])| = 5$ for each v in {A,B,C}. We have thus established that there is no pure NE in the instance of Figure 4.1a. It is easy to extend the above non-existence proof to larger d by replacing selected edges in the instance of Figure 4.1a by multi-hop paths. □



(a) An instance of a contact graph that has no pure NE.

(b) Residual graph when A chooses to secure itself.

(c) Residual graph when A and B choose to secure themselves

Figure 4.1: No pure NE example with nonuniform security costs and infection costs.

In the above non-existence proof, nodes have different security costs and infection costs. We can extend the proof to the case of uniform security costs and infection costs by inserting additional nodes in the proximity of those nodes in the above instance that have lower security costs, as shown in the following lemma.

Lemma 34. *For any fixed d, $1 < d < \infty$, there exists an instance of* GNS(d) *in which* *no pure NE exists.*

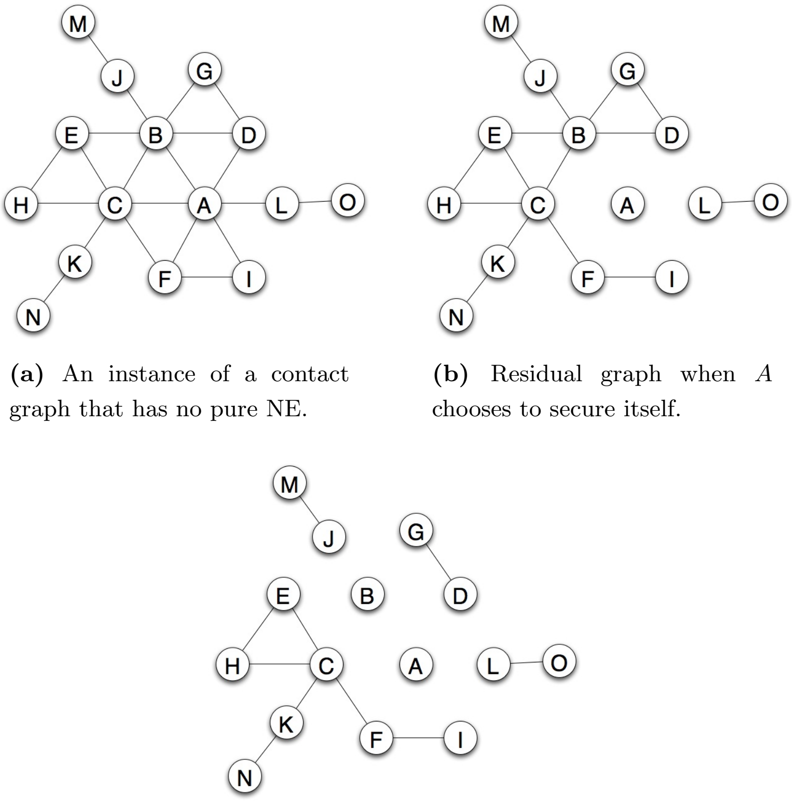
Proof. We first consider the case $d = 2$. Consider the instance defined by the contact graph in Figure 4.2a. $w_v = 1/n$ for all node v. We set the infection cost to be L and security cost to be $C = (10 + \epsilon)L/15$ for all nodes. Thus, in any pure NE $\vec{a}$, node v is secure if and only if $|S_v(\vec{a}[v/0])| \geq 10+\epsilon$. Therefore, nodes D through O are all insecure in any pure NE. We now consider four cases. If all of A, B, and C are insecure in $\vec{a}$, then we do not have a pure NE since $|S_v(\vec{a}[v/0])| = 13$ for each v in {A, B, C}. If exactly one of A, B, or C – say A – is secure, as shown in Figure 4.2b, then B won't change its strategy since $|S_B(\vec{a})| = 10$, but C will change its strategy since $|S_C(\vec{a})| = 11$. If exactly two of A, B, C – say A and B – are secure, as shown in Figure 4.2c, then B will change its strategy since $|S_B(\vec{a}[B/0])| = 10$. Finally, if all three are secure, then none of A, B, or C will stick to its current strategy since $|S_v(\vec{a}[v/0])| = 7$ for each v in {A, B, C}. We have thus established that there is no pure NE in the instance of Figure 4.2a. It is also easy to extend the above non-existence proof to larger d by replacing selected edges in the instance of Figure 4.2a by multi-hop paths. □

We next show that it is, in fact, NP-complete to determine whether a given instance of the generalized network security game with $1 < d < \infty$ has a pure NE. It is easy to argue that the problem is in NP since one can efficiently verify whether a given strategy vector $\vec{a}$ is a pure NE. In the remainder of this section, we focus on the hardness reduction.

Our starting point is the non-existence instance defined in the Lemma 33. We observe that if the security cost of exactly one of the three nodes in {G, H, I}, say G, is reduced so that G always secures itself, then we do have a pure NE in which C secures itself, while A and B are insecure. Thus, if we can control the decision of G through an external input, then we can use the above instance as a gadget which has the property: it has a pure NE if and only if G is secure. We now show how to use this gadget to obtain an NP-hardness reduction.

Theorem 35. *The problem of determining if a* GNS(d) *instance,* $1 < d < \infty$*, has a pure NE is NP-complete.*

Proof. We reduce 3SAT problem to a GNS(2) instance, and show that a given formula ϕ is satisfiable if and only if the corresponding game has a pure NE. The reduction is shown in Figure 4.3. For each variable X in the formula, we create two nodes in the contact graph, X and $\bar{X}$, which are connected to each other. For each literal l in the



(a) An instance of a contact graph that has no pure NE.

(b) Residual graph when A chooses to secure itself.

(c) Residual graph when A and B choose to secure themselves.

Figure 4.2: No pure NE example with uniform security costs and infection costs.

formula, we create a node, and connect it with corresponding variable. For each clause C, we create a gadget, treat node G as clause node, and connect it to its 3 literal nodes. The costs for gadget nodes are as before. The costs of literal nodes are set such that their "threshold" – the number of insecure nodes that can tolerate without securing themselves – is 1. And the threshold for X is set to be $a + 1$ where a is the number of adjacent literal nodes; the threshold for $\bar{X}$ is set to be $b + 1$ where b is the number of adjacent literal nodes. We add padding nodes between edges $(X, \bar{X})$, (X, I), $(\bar{X}, I)$, and (C, I). We set their security costs to be 0, so they always wish to be secure.

We first show if ϕ is satisfiable, then there is a pure NE in this game. For variable node X, if its assignment is true, then make it secure. For literal node I, if its assignment is false, then make it secure. If a clause is true, then make it secure. All the other nodes are insecure. We now argue that the defined strategy vector is a pure

NE. If a variable node X is secure, then all the literal nodes connected to it are not secure, $\bar{X}$ is not secure, while all the literal nodes connected to $\bar{X}$ are secure. Since the formula is satisfiable, all the clause nodes are secure. It is clear that $\bar{X}$ is happy, since its threshold is $b+1$ and X is secure. Similarly X is happy since if it were to be insecure, it will be in a component with size $a+2$ which is bigger than its threshold. All the literal nodes connected to X are happy, because for each of them, the only two adjacent nodes are secure. And all the literal nodes connected to $\bar{X}$ are happy, because if any of them does not secure itself, it will be in a component with size 2, which is bigger than its threshold. All the clause nodes are happy because the formula is satisfiable, at least one of its literal is true, which means at least one of its literal nodes is insecure, hence this clause node has to secure itself because its threshold is 6. And within each gadget, we can make node C to secure itself (together with the nodes D, E, and F) to make all the nodes in the gadget happy. We thus have a pure NE in the game instance.

Next, we argue if the game has a pure NE, then the formula is satisfiable. Suppose we have a pure NE strategy vector $\vec{a}$. For each variable node X, if X is secure, we assign X to be true for the SAT formula; and false otherwise. We know that in any pure NE, the clause node in each gadget has to be secure. Furthermore, exactly only of X and $\bar{X}$ is secure. If X is secure, then $\bar{X}$ and all the literal nodes connected to X have to be insecure, while all the literal nodes connected to $\bar{X}$ have to be secure. Since all the clause nodes are happy, at least one of its literal nodes is not secure, implying that in each clause at least one of the literals is true. This establishes that the formula is satisfiable.

In sum, the formula is satisfiable if and only if the security game has Nash equilibrium. It is easy to see that the above reduction can be carried out in polynomial time, thus yielding the NP-hardness of the problem. □

4.4 Optimizing social welfare: NP-completeness and approximation algorithms

4.4.1 NP-completeness of computing the social optimum

We show that computing the social optimum is NP-complete in GNS(d) games for all d. The result for $d = \infty$ follows from Aspnes et al. [15], even for the special case where all security costs, infection costs, and initial infection probabilities are uniform. We now establish NP-completeness for all $d > 0$.

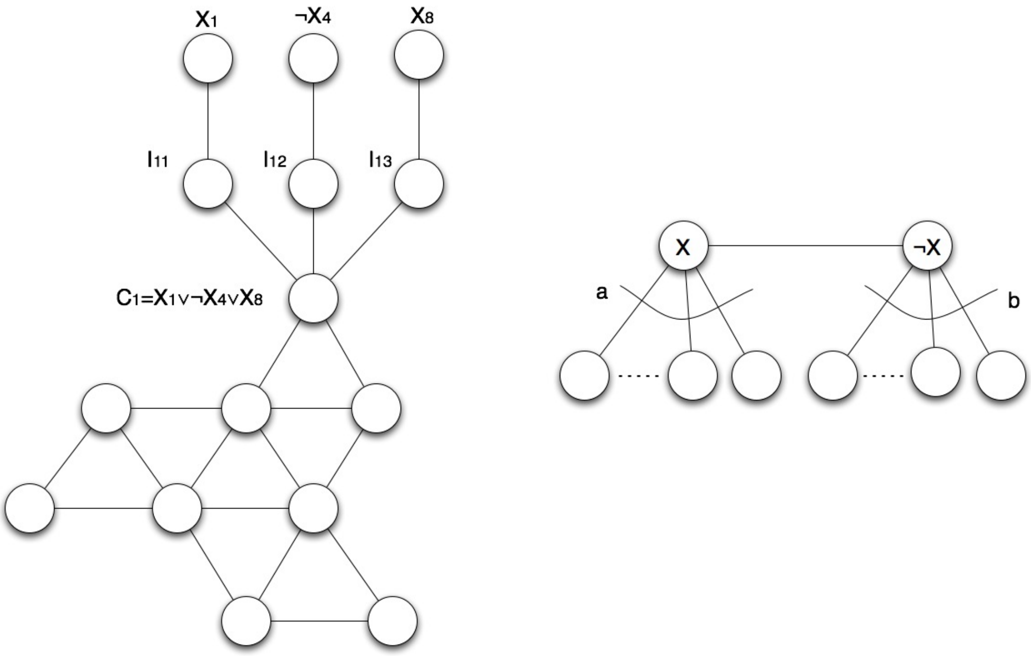


Figure 4.3: Reduction from 3SAT to GNS(d). X_i's refer to variables in the boolean formula. I_{ij} refers to the jth literal in the ith clause. And C_i's refer to the clauses.

Lemma 36. *Computing the social optimum for an instance of* GNS(d) *is NP-complete for all d.*

Proof. We construct a reduction from vertex cover on regular graphs, which is also NP-complete [63]. Consider an instance of vertex cover specified by an r-regular graph $G = (V, E)$. We construct an instance $\mathcal{I}$ of the GNS(d) problem as follows. Let $H = (V', E')$ be a graph obtained by splitting each edge $e = (u, v) \in E$ by $d - 1$ auxiliary nodes $v_{e,1}, \ldots, v_{e,d-1}$, so that $V' = V \cup \cup_{e \in E}\{v_{e,1}, \ldots, v_{e,d-1}\}$, and E' consists of the edges $\cup_{e=(u,v)\in E}\{(u, v_{e,1}), (v, v_{e,d-1}), (v_{e,1}, v_{e,2}), \ldots, (v_{e,d-2}, v_{e,d-1})\}$. For all nodes $v \in V$, let them have the same secure cost C and infection cost L. And we set $C = \frac{L(r(d-1)+1)}{|V'|} + 1$. For each $u \in V' \setminus V$, we have $L_u = 1/|V'|^3$ and $C_u = (C + L)|V'|$. This ensures all nodes in $V' \setminus V$ are insecure, and $\sum_{u \in V' \setminus V} \mathrm{cost}_u(\vec{a}) \leq \epsilon$ for small constant ϵ, for any strategy $\vec{a}$.

Let $B = \{v \in V : a_v = 1\}$ for a pure strategy $\vec{a}$, and let $b = |B|$. It is easy to verify that $\mathrm{cost}(\vec{a}) = \frac{L|V|(r(d-1)+1)}{|V'|} + b + \epsilon + \frac{2L}{|V'|}|\{e = (u, v) : u, v \in V, a_u = a_v = 0\}|$. Therefore, when we set $L > |V| \cdot |V'|$, B is a vertex cover in G of size k, if and only if the social optimum in $\mathcal{I}$ is at most $\frac{L|V|(r(d-1)+1)}{|V'|} + k + \epsilon$. □

For $d = 1$, we also show that while a pure NE always exists, finding the least cost one is NP-complete.

Lemma 37. *Finding the least cost pure NE in a given instance of* GNS(1) *is NP-complete.*

Proof. Our proof is a reduction from Vertex Cover. Let G be an instance of vertex cover. We construct an instance $\mathcal{I}$ of the game in the following manner. We set the contact graph to be $H = (V', E')$ with $V' = V \cup \cup_{i \in V} A(i)$, where the set $A(i) = \{v_{i,1}, \ldots, v_{i,t}\}$, for $t \geq \Delta(G)$, where $\Delta(G)$ is the maximum degree of G. The set E' consists of E along with the edges (i, j), for all $i \in V$ and $j \in A(i)$. The security and infection costs for all nodes in V are identical, C and L, respectively. Set $C = \frac{(t+1)L}{|V'|} + 1$. For nodes in $V' \setminus V$, these corresponding costs are $C' = L'(1 + \epsilon)/|V'|$ and $L' = 1/M$, respectively, where $M \geq |V'|^2 t$. We assume that the initial infection probability distribution is uniform. Therefore, the contribution, $\mathrm{cost}_v(\vec{a})$ of a node $v \in V' \setminus V$ to the total cost $\mathrm{cost}(\vec{a})$ for any strategy vector $\vec{a}$ is at most $\max\{C', 2L'/|V'|\}$, and the total contribution of all such nodes is at most 1. We show that the least cost NE has cost very close to the social optimum.

Let A be a vertex cover for G, with $|A| = a$. Consider the following strategy vector $\vec{a}$: for each $i \in A$, we have $a_i = 1$ and $a_{v_{i,j}} = 0$ for all j, and for $i \notin A$, we have $a_i = 0$

and $a_{v_{i,j}} = 1$ for all j. Following Lemma 28, this vector is a NE because: (i) for each node $i \in A$, there are at least t insecure neighbors (namely, the nodes $v_{i,j}$), (ii) for each $i \notin A$, the number of insecure neighbors is at most $\Delta(G) \leq C|V'|/L$, where $\Delta(G)$ is the maximum degree of G, (iii) if $i \in A$, each node $v_{i,j}$ has no insecure neighbor, and since $C'|V'|/L' = 1 + \epsilon$, such a node won't change its strategy, and (iv) if $i \notin A$, each node $v_{i,j}$ has an insecure neighbor and it will stay being secure. As in the proof of Lemma 36, $\text{cost}(\vec{a}) \leq L + |A| + 1$. Therefore, if G has a vertex cover of size k, the reduced game instance has a pure NE of cost at most $L + k + 1$.

For the converse, let $\vec{a}$ be the strategy vector of a NE, and $A = \{i : a_i = 1\} \cap V$. As in the proof of Lemma 36, $\text{cost}(\vec{a}) = L + |A| + \frac{2L}{|V'|}|\{(u, v) : a_u = a_v = 0,\ u, v \in V\}|$, which implies if A is not a vertex cover for G, $\text{cost}(\vec{a}) > L + |A|$. Therefore, the lemma follows. □

4.4.2 Approximating the social optimum

We describe a general framework to derive approximation algorithms for GNS(d) games for all d. For fixed d, we achieve an approximation ratio of $2d$. For $d = \infty$, we obtain an approximation ratio of $O(\log n)$. Our framework involves the following three steps.

1. Formulate a linear programming relaxation.

2. Let $\mathbf{x}$ be the optimum LP solution. Partially round and filter the variables. Let $\mathbf{x}'$ be the resulting solution.

3. Round the $\mathbf{x}'$ solution appropriately - for constant d, this involves solving a suitable covering problem, while for $d = \infty$ this reduces to a vertex separator problem.

4.4.2.1 An LP Formulation

Let P_{ij}^d denote the set of all simple paths from i to j of length at most d. Let x_v be the indicator variable for node v that is 1 if v is secured. Let y_{ij} be the indicator variable for nodes i and j that is 1 if there is no path $P \in P_{ij}^d$ consisting entirely of insecure nodes. By abuse of notation, for $i = j$, we assume $y_{ii} = 1$ if node i has been secured, i.e., $x_i = 1$. We start with the following integer programming formulation $\mathcal{P}$ of the social optimum.

$$\begin{array}{ll} \min & \sum_v C_v \cdot x_v + \sum_{j \in V} L_j \sum_{i \in V} w_i (1 - y_{ij}) \\ \text{s.t.} & \sum_{v \in p} x_v \geq y_{ij} \; p \in P_{ij}^d \\ & x_v \in \{0,1\} \; \forall v \in V \\ & y_{ij} \in \{0,1\} \; \forall i,j \in V \end{array} \tag{4.1}$$

The objective function can be interpreted in the following manner: the first part corresponds to the cost of securing nodes, and the second part corresponds to the infection cost, which, for node j is L_j times the sum of the probabilities of all nodes that have a path to j of length at most d consisting entirely of insecure nodes. The first constraint says that in order to separate a pair of nodes i and j, we need to secure at least one node in every path $P \in P_{ij}^d$ between these two. For $i = j$, we define the only path P in P_{ij}^d to consist of the node i.

We relax the IP to a linear program (LP) by changing the last two constraints to $0 \leq x_v \leq 1$ and $0 \leq y_{ij} \leq 1$.

4.4.2.2 Solving the LP and partial rounding and filtering

We now perform the following steps.
(1) Solve the LP: for any fixed d, the number of paths of length at most d, $|P_{ij}^d|$ is at most $n^{O(1)}$, and therefore, the above program can be solved in polynomial time. When d is not a constant, the program cannot be written down efficiently but we can solve it in polynomial time using the ellipsoid method. This requires the construction of a polynomial time separation oracle, which, given a candidate solution $(\vec{x}, \vec{y})$, can decide if it is feasible, or finds a constraint that is infeasible. Such a separation oracle can be designed as follows: define the cost of a path to be the sum of the weights x_v of the nodes on the path. For each pair i, j, compute the shortest path from i to j in the graph restricted to the d-hop neighborhood of node i. If this distance exceeds y_{ij}, the constraints for all the paths $p \in P_{ij}^d$ are satisfied. Else, the constraint corresponding to the shortest such path is violated.

Ellipsoid-based methods are, however, expensive to implement in practice. For the case $d = \infty$, we address this drawback by solving an equivalent polynomial-sized LP in which we introduce a “distance variable” for each pair of nodes and replace the

exponentially-many path constraints given in (4.1) with polynomially-many triangle inequality constraints, and linear number of lower bounds on the distances. It is this more compact LP that we solve in our experiments.
(2) Construct a new vector $\vec{y}'$ in the following manner: for each i, j, $y'_{ij} = 0$ if $y_{ij} \leq 1/2$ and $y'_{ij} = 1$ if $y_{ij} > 1/2$. Next, let $x'_v = \min\{2x_v, 1\}$, for all $v \in V$.

4.4.2.3 Final rounding

We now round the vector $\vec{x}'$ to an integral solution. For $d = 1$, it is easy to see that $\vec{x}'$ is already integral, since each constraint only has two variables. We now consider general d. Consider a pair of nodes i and j such that $y'_{ij} = 1$. By constraint (4.1), along every path p of length at most d between i and j, the sum, over $v \in p$, of x'_v is at least 1. It follows that along every such path p, there exists at least one vertex $v \in p$ with $x'_v \geq 1/d$. Consider now the following filtering procedure: if $x'_v \leq 1/d$, we set $x''_v = 0$; otherwise, we set $x''_v = 1$. It is clear that all the constraints of the LP are satisfied, and the cost of $\vec{x''}$ is at most d times the cost of $\vec{x'}$, yielding a final $2d$ approximation.

We finally consider the $d = \infty$ case. In this case, we are left with a minimum weighted vertex multi-cut problem, where we would like to determine the minimum weight of vertices that can separate all the pairs (i, j) for which $y'_{ij} = 1$. The elegant LP rounding algorithm of [68] yields an integral solution for the vertex multi-cut problem, whose cost is $O(\log n)$ times the cost of fractional solution. We can thus find a set X of vertices to secure such that all pairs of vertices for which $y'_{ij} = 1$ are separated and $\sum_{v \in X} C_v$ is at most $O((\log n) \sum_v C_v x'_v)$.

Putting the above analyses together, we have the following.

Theorem 38. *For any fixed d, the social optimum for an instance of* GNS(d) *can be approximated to within a factor of $2d$ in polynomial time. For $d = \infty$, we obtain an $O(\log n)$-approximation to the social optimum, where n is the number of nodes in the contact graph.*

4.5 Experimental results

We now empirically study the properties of NE and the performance of our algorithms. We use two classes of graphs: (i) random geometric graphs formed by distributing n^2 nodes uniformly at random in an $n \times n$ square and add an edge between a pair of nodes if there distance is no more than 1, and (ii) power law graphs generated by preferential attachment process [27]. These two graph classes are very different, with the former being a model for wireless networks, while the latter suited for the Internet [62], World Wide Web [27], and email networks [58]. Also, they have very contrasting properties, e.g., the latter class has larger separators, and we expect to see effects of these differences. We set the infection costs to be identical for every node (this can be done without loss of generality for the pure NE analysis) and the security costs are chosen uniformly at random between 0 and the infection cost.

Our main experimental observations are the following.

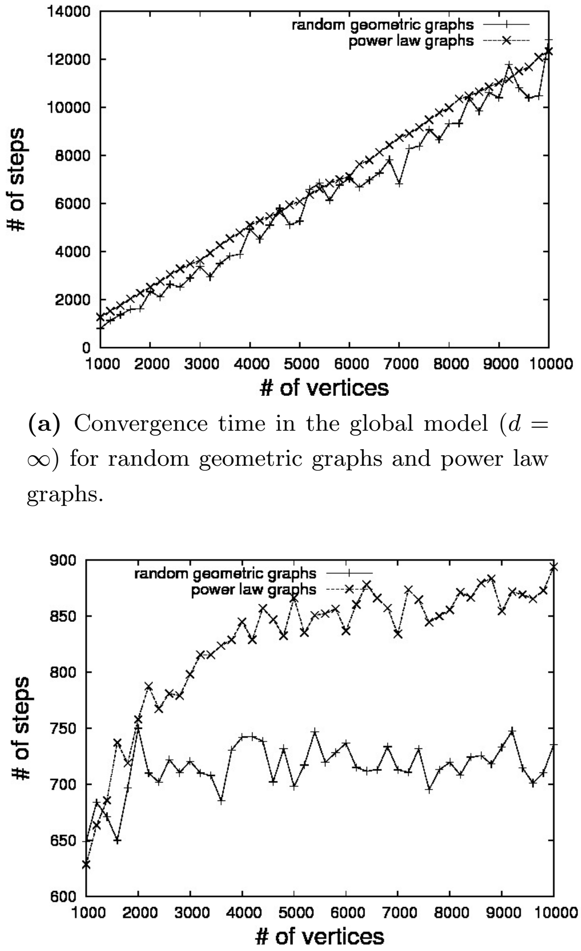
1. *Convergence time for best response strategies*: We find that best response works pretty well in practice. For $d = \infty$, we find the convergence time to be linear in the number of nodes for both graph classes, while it seems to be sub-linear in the case of $d = 1$. For the d-neighborhood model, with $1 < d < \infty$, best response does not converge to a NE quite often, suggesting that even on average, these games do not have NE.

2. *Structural properties of NE and the quality of NE*: We find that high degree nodes tend to be secured in the NE for the local game. Additionally, we find that the cost of NE is very low for $d = 1$ in both the graph classes, but it is somewhat high for $d = \infty$.

3. *Performance of our approximation algorithms for the social optimum*: While we show a worst case bound of $O(\log n)$ for approximating the social optimum (Section 4.4), we find that our algorithms perform much better in practice. For $d = 1$, the approximation bound is very close to 1; while for $d = \infty$, it seems to be a constant.

4.5.1 Convergence times for best response strategies

We implement best response in a round robin fashion on both the graph classes and study the convergence time; note that the results of Section 4.3 imply that this converges to a NE. Figure 4.4a shows that the convergence time of the global model for random geometric and power law graphs grows linearly with the number of nodes. Figure 4.4b shows the corresponding plots for the local model and they seem to grow much slower than in the $d = \infty$ case. Also, for the d-neighborhood model, we find that best response often does not converge to a NE.



(a) Convergence time in the global model ($d = \infty$) for random geometric graphs and power law graphs.

(b) Convergence time in the local model ($d = 1$) for random geometric graphs and power law graphs.

Figure 4.4: Convergence time.

4.5.2 Structural properties of NE

In Figure 4.5, we examine the degrees of secured nodes in the NE computed by best response on power law graph with 5000 vertices, and we find that they tend to be high. In fact, the degree distribution of the secured nodes seems to mirror the overall degree distribution in the graph. We also study the quality of NE in the local and global models. Figure 4.6a and 4.6b show that the cost of NE is very low for the local model in both graph classes. The ratio to optimal value is at most 1.3. In contrast, Figure 4.7a and 4.7b show that this ratio is larger for the global model, about 7 in both graph classes. We note that this ratio is for the case of non-uniform costs; we expect the ratio to be smaller with uniform costs, especially for power-law graphs owing to their high vertex expansion.

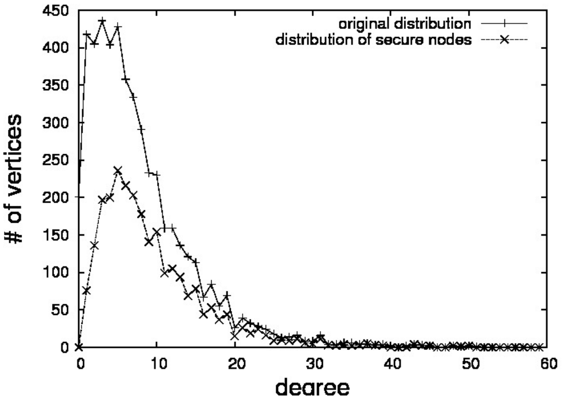
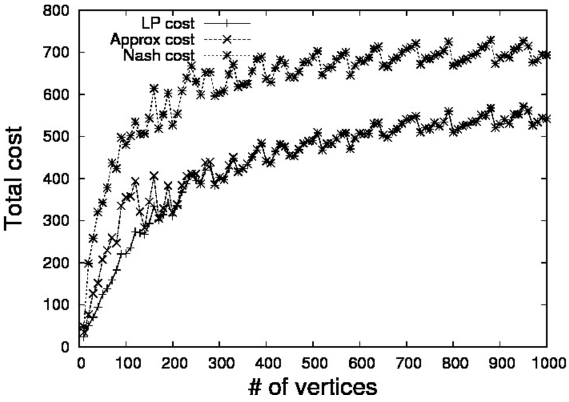


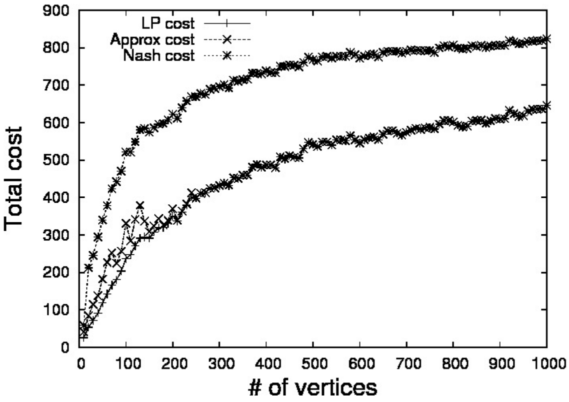
Figure 4.5: Properties of secured nodes in NE in power law graphs.

4.5.3 Empirical performance of approximation algorithms

We now study the empirical performance of the algorithms we design in Section 4.4 for approximating the social optimum. Since computing social optimum is very expensive, we use LP optimal values as lower bound. Figure 4.6a and 4.6b show that our approximation algorithm's cost is almost the same as the LP lower bound for the local model. For the global model, Figure 4.7a and 4.7b show that the approximation algorithm's cost is within a constant of the LP lower bound, in contrast to the worst case $O(\log n)$ bound we prove. Additionally, we observe that our approximation algorithm has a much better guarantee for power law graphs than for random geometric graphs.

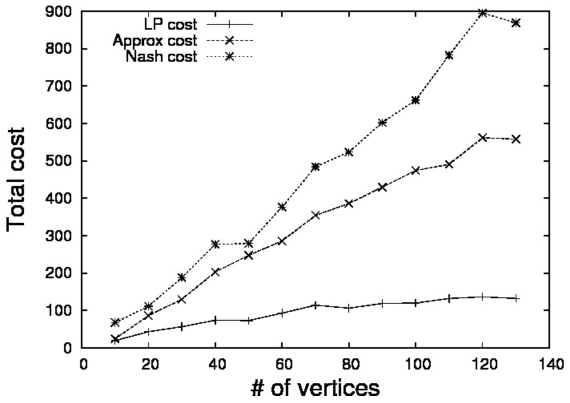


(a) The costs of the LP solution, our approximation algorithm, and the Nash equilibrium computed by best response, for the local model in random geometric graphs.

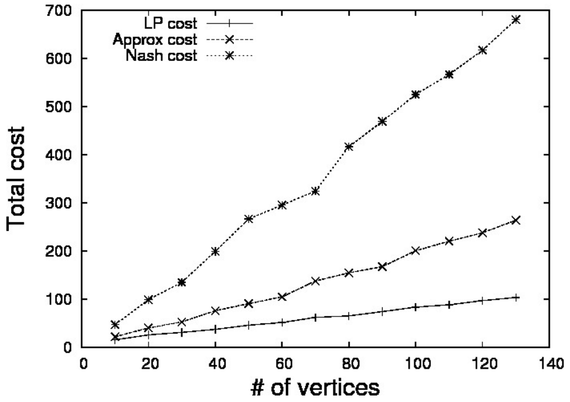


(b) The costs of the LP solution, our approximation algorithm, and the Nash equilibrium computed by best response, for the local model in power law graphs.

Figure 4.6: Costs comparison for the local model.



(a) The costs of the LP solution, our approximation algorithm, and the Nash equilibrium computed by best response, for the global model in random geometric graphs.



(b) The costs of the LP solution, our approximation algorithm, and the Nash equilibrium computed by best response, for the global model in power law graphs.

Figure 4.7: Costs comparison for the global model.

4.6 Conclusion

Non-cooperative games have been recognized as a useful paradigm for studying decentralized network security problems; however, the resources needed for individual decision making are important issues for the implementability of such games. In this paper, we have developed a framework for network security games parametrized by the amount of local information available for individual decision making. We find this parameter plays an important role in the structure of the equilibria, and needs to be taken into account in such analysis.

NE are considered as natural operating configurations in such systems with selfish users. Therefore, ensuring that the system has efficient NE is desirable (equivalently, a low price of anarchy (PoA)) for network planners. Specifically, if the network planner has a limited budget to secure k nodes, an important design problem is to choose a subset of nodes to secure so that the graph restricted to the remaining nodes has low PoA; such a strategy is also referred to as a *Stackelberg* strategy for the network planner [111]. Lemmas 30 and 32, which bound the PoA in terms of the network parameters, suggest natural heuristics to design stackelberg strategies for the network planner. We discuss this briefly below.

In the neighborhood model, Lemma 30 shows that PoA is bounded by $\Delta + 1$. Therefore, given a budget to secure k nodes, the Stackelberg question is to choose a subset of nodes to secure, so that the maximum degree of the residual graph is minimized. An analogous question, dual to this, is the following: for a given target maximum degree Δ', choose the smallest set k of nodes to secure so that the maximum degree in the residual graph is Δ'. Both these versions are NP-complete to solve optimally, but greedy heuristics are likely to perform well. In the global model, Lemma 32 shows that the PoA is bounded by $1/\alpha(\mathcal{G})$. The analogous question of finding an optimal Stackelberg strategy is NP-complete in this case also. We can use the spectral clustering algorithm of [78], which finds an (α, ϵ) clustering of low cost using at most an ϵ fraction of the edges, while ensuring that each cluster has expansion at least α, as a natural heuristic for this problem.

Chapter 5

Controlling negative diffusion in the presence of risk behavior changes

In Chapter 4, we analyzed intervention strategies assuming the behavior of each individual remains the same before and after taking interventions, which is not an accurate assumption in some real world scenarios. Previous studies have shown imperfect interventions and risk behavior changes can lead to perverse outcomes. Thus, in this chapter, we study how to control negative diffusion with the presence of risk behavior changes.

From the results in Chapter 4, we can see that Nash equilibrium may not exist even without risk behavior changes. Using game theory in the presence of risk behavior changes is going to be extremely difficult. In this chapter, we formulate a network-based model and use random graph techniques to understand how risk behavior change in conjunction with failure of prophylactic interventions can lead to perverse outcomes where "less (intervention) is more (effective)". Our model captures the distinction between one- and two-sided risk behavior change. In one-sided situations (e.g. influenza/H1N1) it is sufficient for either individual in an interaction to exhibit risk behavior change whereas in two-sided situations (e.g. AIDS/HIV) it is necessary for both individuals in the interaction to exhibit risk behavior change, for a potential transmission of the disease. A central discovery is that the phenomenon of perversity occurs at differing levels of intervention coverage depending upon the "sidedness" of the interaction. Furthermore, again dependent on the "sidedness," targeting highly connected nodes can be strictly worse than uniformly random interventions at the same level of coverage.

In Section 5.1, we formally define our model. In Section 5.2, we explain our first finding where less intervention can be more effective. In Section 5.3, we explain our second finding where targeted intervention strategy can be worse than random intervention strategy. Section 5.4 backs up our findings with comprehensive simulations. And we conclude this Chapter in Section 5.5.

5.1 Models

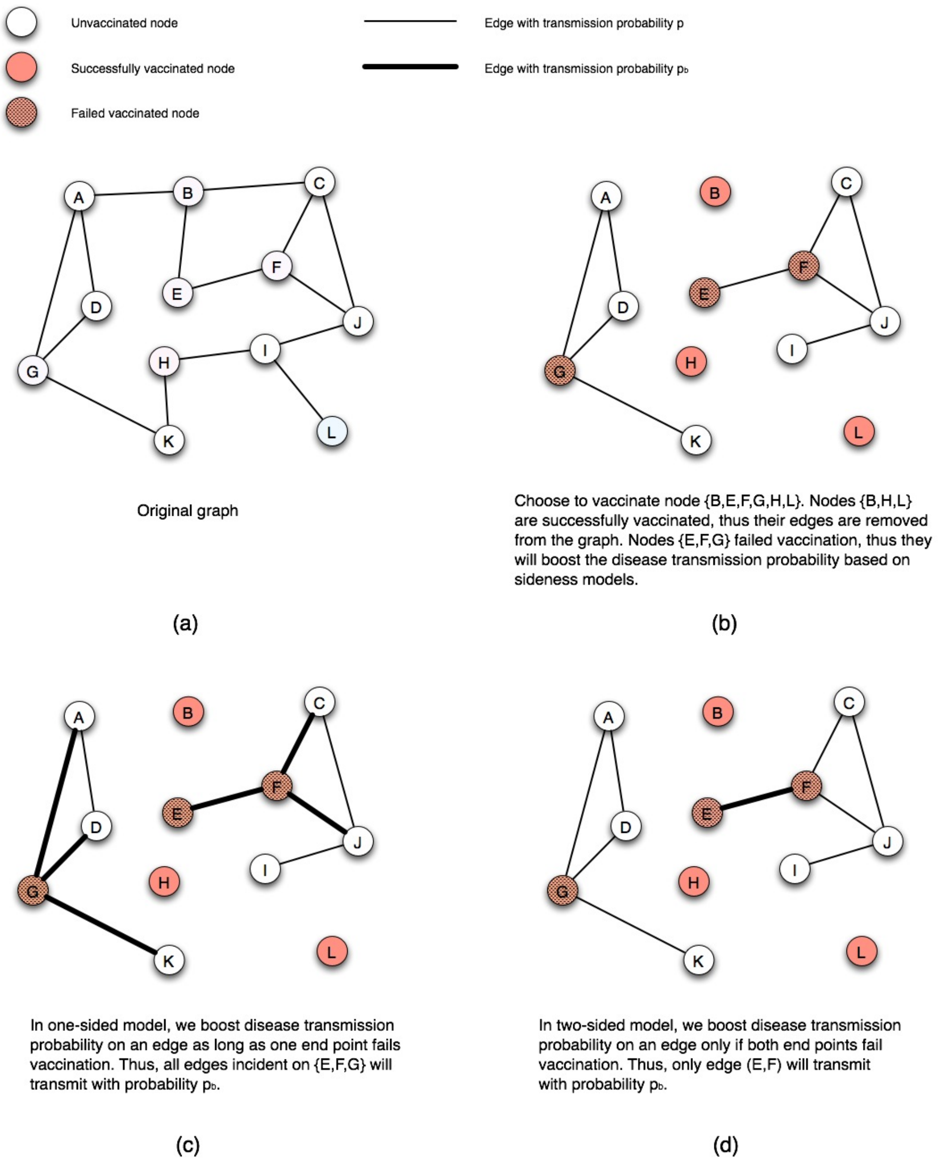
We obtain our results through both analytical techniques and simulations on a range of networks including preferential attachment networks [28] and large synthetic and real-world networks. For our analyses, we adopt the SIR model of epidemics defined on networks. Let $G = (V, E)$ denote an undirected social contact graph, where V denotes a set of people (referred to as nodes henceforth) and $(u, v) \in E$ denotes a contact between nodes u and v (see Figure 5.1(a) for an example). If node u becomes infectious, it will infect each of its susceptible neighbors independently with probability p (referred as *base transmissivity*). Each node in the graph is either vaccinated (e.g., nodes B or F in Figure 5.1(b)) or not (e.g., nodes A or C in Figure 5.1(b)). If a node u is not vaccinated, we label it as UV. The vaccine fails with probability p_f. If a node u's vaccine fails, we label it as VF; otherwise, we label it VS. Both UV and VF nodes are susceptible. We assume that vaccine failure is a stochastic event and that (vaccinated) nodes do not know if (their own) vaccination succeeded or not. If a node with vaccine failure is infected then its risk behavior changes, i.e., it increases its contacts to some of its' neighbors, resulting in *boosted transmissivity* p_m - in the one-sided model a node infects all its susceptible neighbors with boosted transmissivity p_m, while in the two-sided model, it only infects those neighbors with boosted transmissivity p_m that have also had a failed vaccination. In the rest of the paper, we use p_v to denote the probability that a node is vaccinated, under a campaign of uniformly random vaccination.

The disease transmission process is thus defined by the tuple (p, p_m, p_f, p_v) in the following manner: every node is labeled with UV, VS, VF with probability $1 - p_v$, $p_v(1 - p_f)$, and $p_v p_f$, respectively. All nodes labeled VS are removed from the graph. Each edge (u, v) connecting two surviving nodes u and v, is "open" (or retained in the graph, in the language of percolation, which corresponds to disease transmission on this edge), or "closed" (or removed from the graph), with some probability depending on the model - (i) in the one-sided model, edge (u, v) is open with probability p, if both u and v are labeled UV, and is open with probability p_m if one of u and v is labeled VF; (ii) in the two-sided model, edge (u, v) is open with probability p, unless both u and v are labeled VF. Following the well known correspondence between bond percolation and disease transmission, the connected component containing a specific node u is the (random) subset of nodes infected, if the disease starts at u. If the components resulting from one random instance of the above stochastic process are $C_1, C_2, \ldots, C_k$, then $\sum_i |C_i|^2/n$ denotes the expected outbreak size of the disease starting from a random initial node. In our analysis, we use this as a measure of *epidemic severity.*

5.2 Perversity and sidedness

We first report on our finding that both one-sided and two-sided behavior changes can lead to perverse outcomes (less vaccination is more effective) across a wide range of contact networks. One-sided behavior change leads to perverse outcomes at low levels of intervention, in which the epidemic severity increases with p_v, up to a point, as shown in Figure 5.2, 5.3, and 5.4. Two-sided behavior change leads to perverse outcomes at high levels of intervention, in which the epidemic severity starts increasing beyond a threshold value of p_v. We mathematically establish the phenomena of perversity and non-monotonicity for graphs generated according to the Erdös-Renyi model [108], denoted by $G(n, p)$, in which each edge between a pair of nodes is chosen independently with probability p. We prove rigorously that there exist p, p_m, and p_f, such that (i) in the one-sided model, it almost surely holds that the epidemic severity is $o(n)$ for both $p_v = 0$ and $p_v = 1$, yet $\Theta(n)$ for some p_v in $(0, 1)$; (ii) in the two-sided model, the epidemic severity is $\Theta(n)$ for both $p_v = 0$ and $p_v = 1$, yet $o(n)$ for some p_v in $(0, 1)$. This implies that there is a choice of parameters (which turns out to be be quite broad), such that as the vaccinated fraction p_v is varied, the epidemic severity shows a non-monotone behavior.

Theorem 39. *For the Erdös-Rényi random graph model $G(n,p)$, there exist p, p_m, and p_f, such that (i) in the one-sided model, it almost surely holds that the epidemic severity is $o(n)$ for both $p_v = 0$ and $p_v = 1$, yet $\Theta(n)$ for some p_v in $(0, 1)$; (ii) in the*


Unvaccinated node
Successfully vaccinated node
Failed vaccinated node
Edge with transmission probability p
Edge with transmission probability pb
A
B
C
D
E
F
G
H
I
J
K
L
(a)
(b)
(c)
(d)

Original graph

Choose to vaccinate node {B,E,F,G,H,L}. Nodes {B,H,L} are successfully vaccinated, thus their edges are removed from the graph. Nodes {E,F,G} failed vaccination, thus they will boost the disease transmission probability based on sideness models.

In one-sided model, we boost disease transmission probability on an edge as long as one end point fails vaccination. Thus, all edges incident on {E,F,G} will transmit with probability p_b.

In two-sided model, we boost disease transmission probability on an edge only if both end points fail vaccination. Thus, only edge (E,F) will transmit with probability p_b.

Figure 5.1: Sidedness of risk behavior change: the one-sided and two-sided models.

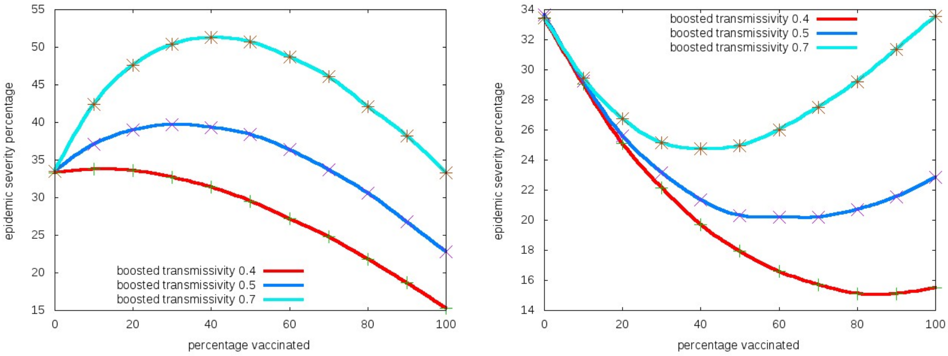


Figure 5.2: Epidemic severity with different boosted transmissivities in one-sided (left) and two-sided (right) risk behavior models. x-axis is the percentage of nodes taking interventions, and y-axis is the expected percentage of nodes getting infected. $p = 0.25$ and $p_s = 0.35$.

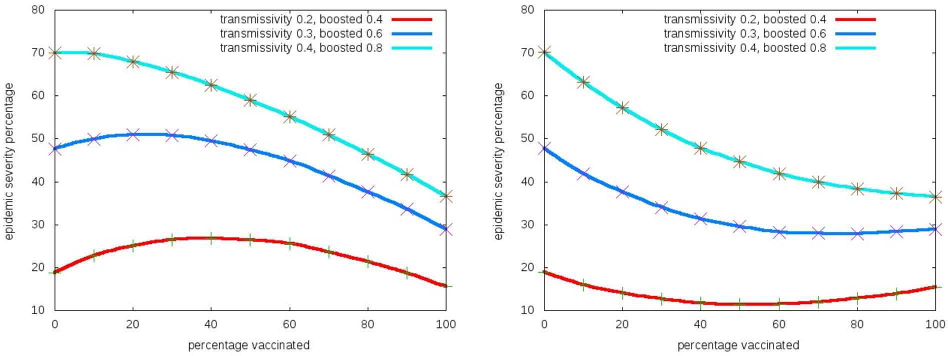


Figure 5.3: Epidemic severity with different transmissivities in one-sided (left) and two-sided (right) risk behavior models. x-axis is the percentage of nodes taking interventions, and y-axis is the expected percentage of nodes getting infected. $p_s = 0.35$, and $p_m = 2p$.

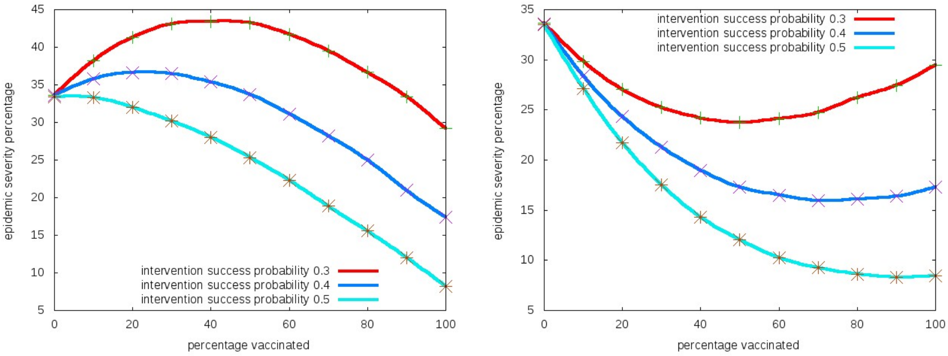


Figure 5.4: Epidemic severity with different intervention success probabilities in one-sided (left) and two-sided (right) risk behavior models. x-axis is the percentage of nodes taking interventions, and y-axis is the expected percentage of nodes getting infected. $p = 0.25$, and $p_m = 0.5$.

two-sided model, the epidemic severity is $\Theta(n)$ *for both* $p_v = 0$ *and* $p_v = 1$*, yet* $o(n)$ *for some* p_v *in* $(0, 1)$.

We give a brief sketch of our proof, which is based on recent results of Söderberg [120] and Bollobás et al [39] on heterogeneous random graphs. We refer the reader to supplementary information for details. Consider the model of heterogeneous random graphs denoted by $\mathcal{G}(N, K, \mathbf{r}, \mathbf{c})$, where (i) K is a positive integer, (ii) $\mathbf{r} = \{r_1, \ldots, r_K\}$ is a probability vector, (iii) $\mathbf{c} = (c_{ij})$ is a $K \times K$ matrix, (iv) each node $j = 1, \ldots, N$, is assigned a type $i \in \{1, \ldots, K\}$ with probability r_i, and (v) each pair of nodes i, j are connected by an edge with probability $p(i, j) = c_{ij}/N$. Söderberg [120] and Bollobás et al. [39] established the following: (i) if the eigenvalues of the matrix $\{c_{ij} r_j\}$ are all less than 1, it is sub-critical (i.e., has no giant component), and (ii) if some eigenvalue is larger than 1, it is super-critical (i.e., has a giant component) with asymptotically $r_i(1 - f_i)N$ nodes of type i, where f_i satisfies the coupled set of equations: $f_i = exp\left(\sum_j c_{ij} r_j (f_j - 1)\right)$. We show that if the contact network is generated by the Erdos-Renyi model $G(n, c/n)$, then the disease transmission process produces a heterogeneous random graph with the eigenvalue characteristic equation given by

$$-\lambda(\lambda^2 - (c(1 - p_v) + p_m c p_v p_f)\lambda + c^2(1 - p_v)p_m p_v p_f - c^2(1 - p_v)p_v p_f) = 0.$$

We show the existence of parameters p_v, p_f, p_m, and c such that the absolute value of every eigenvalue is smaller than 1.

We find the phenomenon of perversity exists in a broad class of graphs, and in order to formally prove its widespread occurrence, we consider *locally finite graphs*, which have been widely studied in percolation theory (e.g., see [38]). Locally finite graphs include infinite graphs in which each node has bounded degree. Using techniques from percolation theory, we prove that in every locally finite graph G, there exist p, p_m, and p_f, such that: (i) the epidemic severity is finite for both $p_v = 0$ and $p_v = 1$, yet infinite for some p_v in $(0, 1)$ in the one-sided model; (ii) the epidemic severity is infinite for both $p_v = 0$ and $p_v = 1$, yet finite for some p_v in $(0, 1)$ in the two-sided model. This result provides strong evidence of the universality of the phenomenon. As such it begs for a natural and intuitive explanation. Our best structural understanding at this point is that this is the consequence of two competing tensions – vaccine success that serves to contain the spread and risky behavior that, exacerbated by vaccine failure, serves to boost the contagion. In the one-sided situation since it is sufficient for infection spread to have just the one party in an interaction exhibiting risky behavior we see perversity manifesting itself at low levels of vaccination. Whereas, in the two-sided situation since it is necessary for both the interacting parties to exhibit risky behavior we see perversity manifesting itself only at high vaccination levels which is a prerequisite for a non-trivial fraction of parties with failed vaccines to exist.

Theorem 40. *For every locally-finite infinite graph G, there exist p, p_m, and p_f, such that: (i) the epidemic severity is finite for both $p_v = 0$ and $p_v = 1$, yet infinite for some p_v in $(0, 1)$ in the one-sided model; (ii) the epidemic severity is infinite for both $p_v = 0$ and $p_v = 1$, yet finite for some p_v in $(0, 1)$ in the two-sided model.*

The phenomenon of non-monotonicity and its dependence on sidedness that we have identified occurs across a wide range of network models.

5.2.1 Proof of Theorem 39

In this section, we give formal proofs of perversity and non-monotonicity in Erdös-Rényi random graphs. We have observed the non-monotonicity is pervasive in wide range of contact graphs, including scale-free graphs, Erdös-Rényi graphs, and other synthetic or real world graphs. Theorem 39 gives the rigorous proof of one-sided model and two-sided model for Erdös-Rényi random graphs.

Lemma 41. *Given a complete graph as the contact network, for intervention with any success probability p_s, there exists parameter set p, p_m, p_v, such that there is non-monotonicity in two-sided risk behavior model.*

Proof. When nobody takes interventions, there are n nodes in the graph, and the disease transmission probability between each pair of nodes is $p = c/n$, where $c > 1$. By [60], there is a giant connected component with high probability (size of the connected component is $\Theta(n)$).

When everybody takes interventions, $p_s n + o(n)$ nodes will have successful interventions with high probability, and thus removed from the graph. The remaining $(1 - p_s)n + o(n)$ nodes will all exhibit risk behavior changes. Thus, the disease transmission probability between each pair of nodes is $p_m = c'/(1 - p_s)n$, where $c' > 1$. By [60], there is a giant connected component with high probability (size of the connected component is $\Theta(n)$).

Now we are going to show there exists a p_v, such that, if we apply interventions to each node independently with probability p_v, the epidemic severity will be $o(n)$ with high probability. Let A be the set of nodes that haven't taken interventions, B be the set of nodes that have taken interventions but failed, and C be the set of nodes that have taken interventions and succeeded. $r_A = 1 - p_v$ represents the probability of a random node being in set A, $r_B = p_v (1 - p_s)$ represents the probability of a random node being in set B, and $r_C = p_v p_s$ represents the probability of a random node being in set C. In the two-sided model, disease transmits with probability p_m between nodes in set C, and p otherwise. Set $a = c$ and $b = c' / (1 - p_s)$. Let

$$M = \begin{pmatrix} ar_A & ar_B & 0 \\ ar_A & br_B & 0 \\ 0 & 0 & 0 \end{pmatrix}$$

This yields a model of inhomogeneous random graphs with 3 types of vertices (A, B, and C). By [120] Theorem 1, if all the eigenvalues of M are less than 1 in absolute value, then the size of the largest connected component is $o(n)$. Let λ be the eigenvalues of M.

$$\begin{aligned} \det(M - \lambda I) &= \det \begin{bmatrix} ar_A - \lambda & ar_B & 0 \\ ar_A & br_B - \lambda & 0 \\ 0 & 0 & -\lambda \end{bmatrix} \\ &= -\lambda \left((ar_A - \lambda)(br_B - \lambda) - a^2 r_A r_B \right) \\ &= -\lambda \left(\lambda^2 - (ar_A + br_B)\lambda + abr_A r_B - a^2 r_A r_B \right) \end{aligned}$$

Solving $\det(M-\lambda)=0$, we have

$$\lambda_1 = \frac{(ar_A + br_B) + \sqrt{\Delta}}{2}, \lambda_2 = \frac{(ar_A + br_B) - \sqrt{\Delta}}{2}, \lambda_3 = 0$$

where $\Delta = (ar_A - br_B)^2 + 4a^2 r_A r_B$. Since $|\lambda_3| \leq |\lambda_2| \leq |\lambda_1|$, it is sufficient to show there exists a set of parameters that yields $|\lambda_1| < 1$. Set $c' = c$.

$$\begin{aligned} |\lambda_1| &= \frac{(ar_A + br_B) + \sqrt{(ar_A - br_B)^2 + 4a^2 r_A r_B}}{2} \\ &= \frac{c(1-p_v) + c'p_v + \sqrt{(c(1-p_v) - c'p_v)^2 + 4c^2 p_v (1-p_v)(1-p_s)}}{2} \\ &= c\frac{(1-p_v) + p_v + \sqrt{((1-p_v) - p_v)^2 + 4p_v(1-p_v)(1-p_s)}}{2} \\ &= c\frac{1 + \sqrt{((1-p_v) + p_v)^2 - 4p_s p_v (1-p_v)}}{2} \\ &= c\frac{1 + \sqrt{1 - 4p_s p_v (1-p_v)}}{2} \end{aligned}$$

When $0 < p_v < 1$, $\frac{1+\sqrt{1-4p_s p_v(1-p_v)}}{2}$ is a constant smaller than 1. We can find $c > 1$ that satisfies $c\frac{1+\sqrt{1-4p_s p_v(1-p_v)}}{2} < 1$. Thus, for intervention with success probability p_s, there exist parameters p_v, p, and p_m, such that the epidemic size is $o(n)$. This completes our proof of this lemma. □

Lemma 42. *Given a complete graph as the contact network, for intervention with any success probability p_s, there exists parameter set p, p_m, p_v, such that there is non-monotonicity in one-sided risk behavior model.*

Proof. When nobody takes interventions, there are n nodes in the graph, and the disease transmission probability between each pair of nodes is $p = c/n$, where $c < 1$. By [60], the size of the largest connected component is $O(\log n)$ with high probability.

When everybody takes interventions, $p_s n + o(n)$ nodes will have successful interventions with high probability, and thus removed from the graph. The remaining $(1-p_s)n + o(n)$ nodes will exhibit risk behavior changes. Thus, the disease transmission probability between each pair of nodes is $p_m = c'/(1-p_s)n$, where $c' < 1$. By [60], the size of the largest connected component is $O(\log n)$ with high probability.

Now we are going to show there exists a p_v, such that, if we apply interventions to each node independently with probability p_v, the epidemic severity will be $\Theta(n)$ with high probability. Let A be the set of nodes that haven't taken interventions, B be the set of nodes that have taken interventions but failed, and C be the set of nodes that have taken interventions and succeeded. $r_A = 1 - p_v$ represents the probability of a random node being in set A, $r_B = p_v\left(1 - p_s\right)$ represents the probability of a random node being in set B, and $r_C = p_v p_s$ represents the probability of a random node being in set C. In the one-sided model, disease transmit with probability p between nodes in set A, and p_m otherwise. Set $a = c$ and $b = c' / \left(1 - p_s\right)$. Let

$$M = \begin{pmatrix} ar_A & br_B & 0 \\ br_A & br_B & 0 \\ 0 & 0 & 0 \end{pmatrix}$$

This yields a model of inhomogeneous random graphs with 3 types of vertices (A, B, and C). By [120] Theorem 1, if some eigenvalue of M is larger than 1, then the size of the largest connected component is $\Theta\left(n\right)$. Let λ be the eigenvalues of M.

$$\begin{aligned} \det\left(M - \lambda I\right) &= \det \begin{bmatrix} ar_A - \lambda & br_B & 0 \\ br_A & br_B - \lambda & 0 \\ 0 & 0 & -\lambda \end{bmatrix} \\ &= -\lambda\left(\left(ar_A - \lambda\right)\left(br_B - \lambda\right) - b^2 r_A r_B\right) \\ &= -\lambda\left(\lambda^2 - \left(ar_A + br_B\right)\lambda + abr_A r_B - b^2 r_A r_B\right) \end{aligned}$$

Solving $\det\left(M - \lambda\right) = 0$, we have

$$\lambda_1 = \frac{\left(ar_A + br_B\right) + \sqrt{\Delta}}{2}, \lambda_2 = \frac{\left(ar_A + br_B\right) - \sqrt{\Delta}}{2}, \lambda_3 = 0$$

where $\Delta = \left(ar_A - br_B\right)^2 + 4b^2 r_A r_B$. Since $|\lambda_1| \geq |\lambda_2| \geq |\lambda_3|$, it is sufficient to show there exists a set of parameters that yields $|\lambda_1| > 1$. Let $c = c'$.

$$\begin{aligned}
|\lambda_1| &= \frac{(ar_A + br_B) + \sqrt{(ar_A - br_B)^2 + 4b^2 r_A r_B}}{2} \\
&= \frac{c\,(1-p_v) + c'p_v + \sqrt{(c\,(1-p_v) - c'p_v)^2 + 4c'^2 \frac{p_v(1-p_v)}{1-p_s}}}{2} \\
&= c\frac{(1-p_v) + p_v + \sqrt{((1-p_v) - p_v)^2 + 4\frac{p_v(1-p_v)}{1-p_s}}}{2} \\
&= c\frac{1 + \sqrt{((1-p_v) - p_v)^2 + 4p_v\,(1-p_v) - 4p_v\,(1-p_v) + 4\frac{p_v(1-p_v)}{1-p_s}}}{2} \\
&= c\frac{1 + \sqrt{((1-p_v) + p_v)^2 + 4p_v\,(1-p_v)\left(\frac{1}{1-p_s} - 1\right)}}{2} \\
&= c\frac{1 + \sqrt{1 + 4p_v\,(1-p_v)\,\frac{p_s}{1-p_s}}}{2}
\end{aligned}$$

When $0 < p_v < 1$, $\frac{1+\sqrt{1+4p_v(1-p_v)\frac{p_s}{1-p_s}}}{2}$ is a constant greater than 1. We can find $c < 1$ that satisfies $c\frac{1+\sqrt{1+4p_v(1-p_v)\frac{p_s}{1-p_s}}}{2} > 1$. Thus, for vaccination with success probability p_s, there exist parameters p_v, p, and p_m, such that the epidemic size is $\Theta\,(n)$. This completes our proof of this lemma. □

Now we can show the proof of Theorem 39 as follows.

Proof of Theorem 39. We claim the disease transmission process on Erdös-Rényi random graph $G(n, p^*)$ with parameter set (p, p_m, p_s, p_v) is the same as the disease transmission process on a complete graph with parameter set (p^*p, p^*p_m, p_s, p_v). It's simply because the edge between each pair of nodes "opens" with the same probability in both random processes. Thus, for any disease transmission process on Erdös-Rényi random graph, we can reduce it to the corresponding process on a complete graph. Then by Lemma 41 and 42 we can conclude the statement of this theorem holds. □

5.3 Randomized vs. targeted vaccinations

We next report on our finding that targeted vaccination can be strictly worse than random vaccination for some level of vaccine coverage, and this phenomenon occurs both for one-sided as well as two-sided behavior change (as shown in Figure 5.5).

In the literature it has been observed that targeting highly connected individuals for vaccination lead to better outcomes as opposed to random coverage [127, 55, 35]. Our finding adds nuance to the existing results when risky behavior is taken into account. This counterintuitive phenomenon can also be explained by the tug of war between successful vaccination and risky behavior. If the effect of risky behavior is dominant then one would expect that targeted vaccination ends up being worse than random coverage since it is the targeted high-degree individuals that are the most responsible for creating additional contagion. And, in fact the evidence supports this explanation in that we see targeted coverage being inferior to random coverage at low levels of vaccination in the one-sided case but at high levels in the two-sided case.

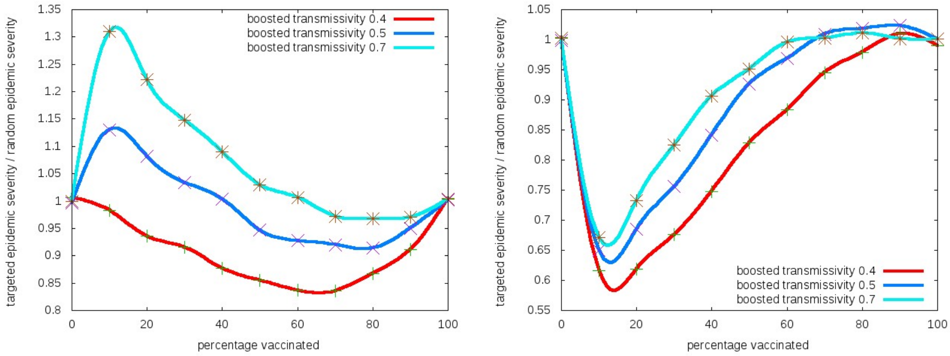


Figure 5.5: Epidemic severity comparison of random and targeted intervention strategies in one-sided (left) and two-sided (right) risk behavior models. x-axis is the percentage of nodes taking interventions, and y-axis is the ratio of the epidemic severity in targeted intervention strategy and the epidemic severity in random intervention strategy. $p = 0.25$, and $p_s = 0.35$.

5.4 Simulations

In order to validate our findings, we carried out comprehensive simulations over a wide range of networks, listed in Table 5.1.

The disease transmission is a random process, defined by the parameter set (p, p_m, p_f, p_v). If a node u becomes infectious, it will infect each of its susceptible neighbors independently with probability p, referred as base transmissivity. Each node in the graph is either vaccinated or not. If a node u is not vaccinated, we label it as UV. If a node u's

Table 5.1: Descriptions of the networks used in the paper. For each network we show its type, name, number of nodes n and edges m.

name		n	m	description
Human contact	NewRiverValley [29]	74,375	1,888,833	Synthetic human contact network for New River Valley county in Virginia.
Social communication	Enron mail [87, 3]	36,691	367,666	Email communication network in a company.
Peer-to-peer network	Gnutella [116, 4]	10,876	39,994	Gnutella peer-to-peer file sharing network from August 2002
Random graphs	Preferential attachment [28]	100,000	300,000	Generated using Python NetworkX library.
	Erdös and Rényi [60]	100,000	5,000,000	

vaccine fails, we label it as VF. Otherwise, we label it VS. Both UV and VF nodes are susceptible. If a node with vaccine failure is infected then its risk behavior changes, resulting in boosted transmissivity p_m. In the one-sided model a node infects all its susceptible neighbors with boosted transmissivity p_m, while in the two-sided model it only infects those neighbors with boosted transmissivity p_m that have also had a failed vaccination. Parameter p_v denotes the probability that a node is vaccinated.

In our simulation, every node is labeled with UV, VS, VF with probability $1-p_v$, $p_v(1-p_f)$, and $p_v p_f$, respectively. All nodes labeled VS are removed from the graph. Each edge (u,v) connecting two surviving nodes u and v is "open", which corresponds to disease transmission on this edge, or "close" with some probability depending on the model - (i) in the one-sided model, edge (u,v) is open with probability p if both u and v are labeled UV, and is open with probability p_m if one of u and v is labeled VF; (ii) in the two-sided model, edge (u,v) is open with probability p, unless both u and v are labeled VF. The closed edges are removed from the graph. In the residual graph, the connected component containing a specific node u is the (random) subset of nodes infected, if the disease starts at u. Let $C_1, C_2, \ldots, C_k$ be the resulting connected components, then $\sum_i |C_i|^2/n$ denotes the expected outbreak size of the disease starting from a random initial node, which we referred as epidemic severity. Since the disease transmission is a random process, for a fixed parameter set we run the simulation for 10 iterations, and take the average value of the epidemic severity. We varified that the epidemic severity is tightly concentrated around the mean, thus the average value of the epidemic severity is a good measure.

We want to confirm our findings: (i) both one-sided and two-sided behavior changes can lead to perverse outcomes (less vaccination is more effective, more precisely, as the vaccinated fraction p_v is varied, the epidemic severity shows a non-monotone behavior); (ii) in both one-sided and two-sided behavior changes, targeted vaccination can be strictly worse than random vaccination for some level of vaccine coverage. For each graph, we run simulations over wide range of parameter set (p, p_m, p_f, p_v), and generate the following 4 sets of plots to validate our findings.

- First set of plots shows how the change of boosted transmissivity will affect the perverse outcomes, as shown in Figure 5.6, 5.10, 5.14, and 5.18. The x-axis is p_v (percentage of vaccinated population) and the y-axis is the epidemic severity (expected percentage of nodes getting infected). We fix the base transmissivity p and the vaccination success probability p_s, then plot the curves for different boosted transmissivity.

- Second set of plots shows how the change of base transmissivity will affect the perverse outcomes, as shown in Figure 5.7, 5.11, 5.15, and 5.19. The x-axis is p_v and the y-axis is the epidemic severity. We fix the vaccination success probability p_s and keep the boosted transmissivity p_m twice the base transmissivity p (i.e. $p_m = 2p$), then plot the curves for different base transmissivity.

- Third set of plots shows how the change of vaccination success probability will affect the perverse outcomes, as shown in Figure 5.8, 5.12, 5.16, and 5.20. The x-axis is p_v and the y-axis is the epidemic severity. We fix the base transmissivity p and the boosted transmissivity p_m, then plot the curves for different vaccination success probability.

- Fourth set of plots shows the finding that targeted vaccination can be strictly worse than random vaccination, as shown in Figure 5.9, 5.13, 5.17, and 5.21. The x-axis is p_v and the y-axis is the ratio between the epidemic severity under targeted vaccination strategy and the epidemic severity under random vaccination strategy. If y value is bigger than 1, it means targeted strategy is worse than random strategy. We fix the base transmissivity p and the vaccination success probability p_s, then plot the curves for different boosted transmissivity.

In order to capture real disease transmission through simulations, we find typical values of R_0, the basic reproduction number, for many diseases such as influenza and HIV [65, 49, 125]. Then, we devide R_0 by the average degree of the graph, and use it as the base transmissivity p. For vaccination success probability, we use the efficacy for real vaccines [61, 70, 1].

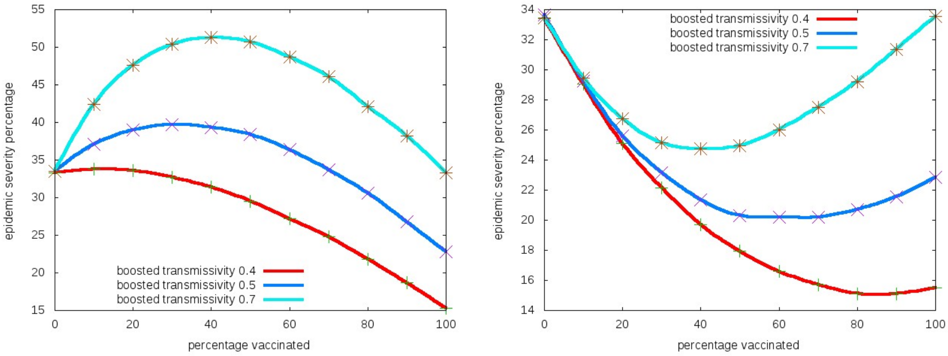


Figure 5.6: Epidemic severity with different boosted transmissivities in one-sided (left) and two-sided (right) risk behavior models. x-axis is the percentage of nodes taking interventions, and y-axis is the expected percentage of nodes getting infected. $p = 0.25$ and $p_s = 0.35$.

5.5 Conclusion

In conclusion, risk behavior change in conjunction with failure of prophylactic interventions can have perverse non-monotone effects on the spread of diseases. This study has explicitly identified sidedness as an attribute of risk behavior change that needs to be taken into account in public policies for vaccinations and antiviral treatments. For one-sided risk behavior change, it is imperative to have sufficiently high levels of coverage, while two-sided situations require both high coverage as well as programs aimed at reducing risky behavior. Our results echo the central premise of Blower-McLean that the development of efficacious prophylactic treatments and increasing their coverage need to go hand in hand with behavioral intervention strategies. These issues need to

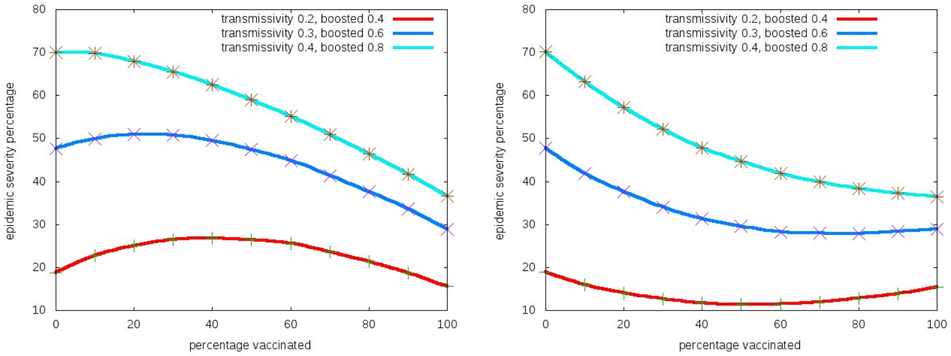


Figure 5.7: Epidemic severity with different transmissivities in one-sided (left) and two-sided (right) risk behavior models. x-axis is the percentage of nodes taking interventions, and y-axis is the expected percentage of nodes getting infected. $p_s = 0.35$, and $p_m = 2p$.

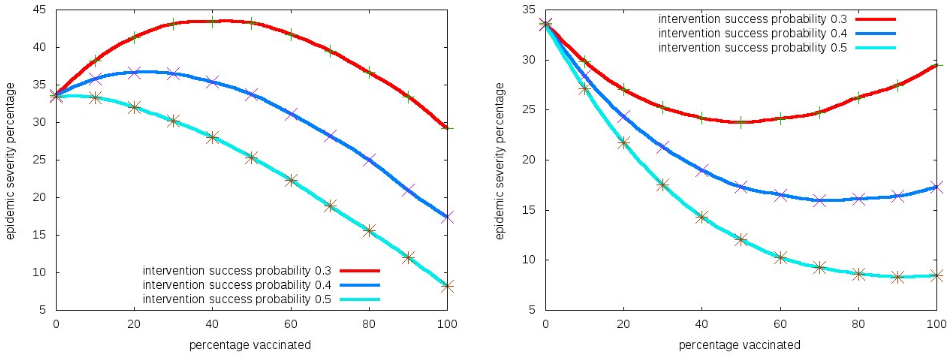


Figure 5.8: Epidemic severity with different intervention success probabilities in one-sided (left) and two-sided (right) risk behavior models. x-axis is the percentage of nodes taking interventions, and y-axis is the expected percentage of nodes getting infected. $p = 0.25$, and $p_m = 0.5$.

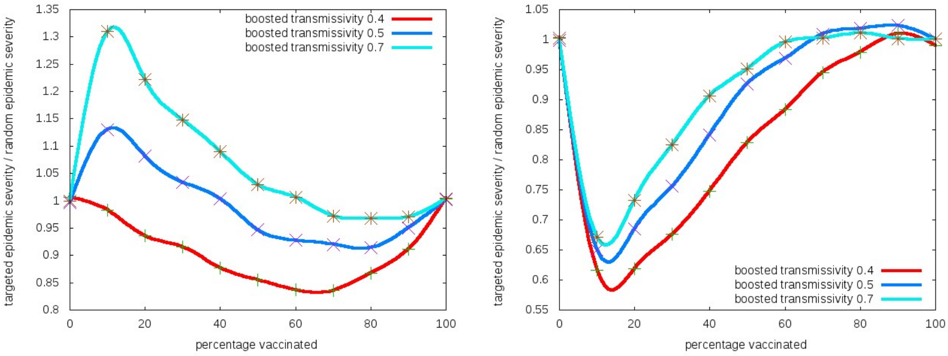


Figure 5.9: Epidemic severity comparison of random and targeted intervention strategies in one-sided (left) and two-sided (right) risk behavior models. x-axis is the percentage of nodes taking interventions, and y-axis is the ratio of the epidemic severity in targeted intervention strategy and the epidemic severity in random intervention strategy. $p = 0.25$, and $p_s = 0.35$.

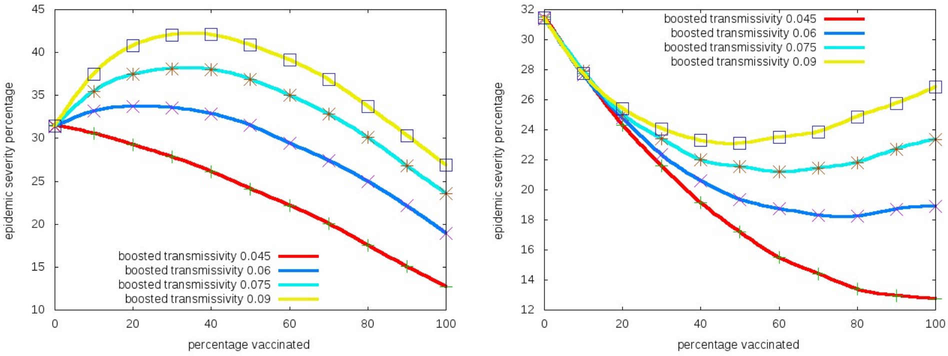


Figure 5.10: Epidemic severity with different boosted transmissivities in one-sided (left) and two-sided (right) risk behavior models. x-axis is the percentage of nodes taking interventions, and y-axis is the expected percentage of nodes getting infected. $p = 0.03$ and $p_s = 0.35$.

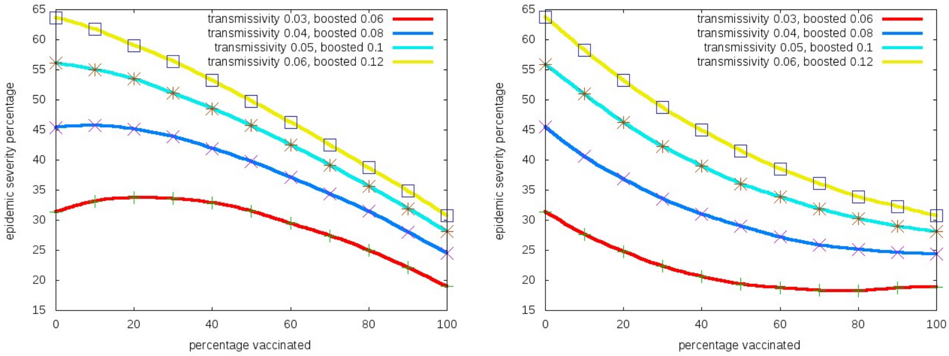


Figure 5.11: Epidemic severity with different transmissivities in one-sided (left) and two-sided (right) risk behavior models. x-axis is the percentage of nodes taking interventions, and y-axis is the expected percentage of nodes getting infected. $p_s = 0.35$, and $p_m = 2p$.

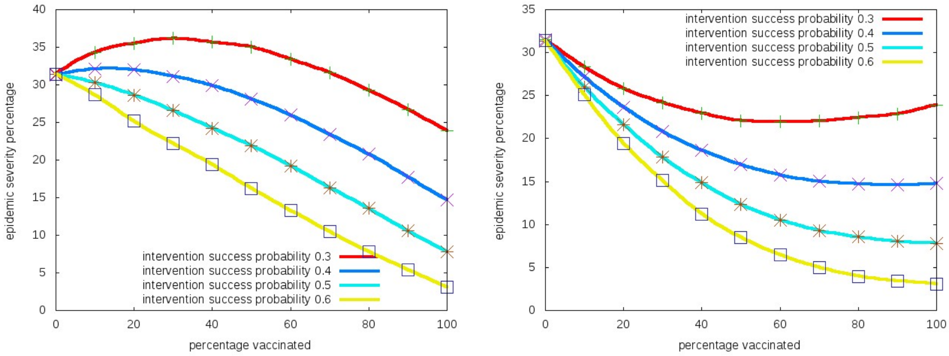


Figure 5.12: Epidemic severity with different intervention success probabilities in one-sided (left) and two-sided (right) risk behavior models. x-axis is the percentage of nodes taking interventions, and y-axis is the expected percentage of nodes getting infected. $p = 0.03$, and $p_m = 0.06$.

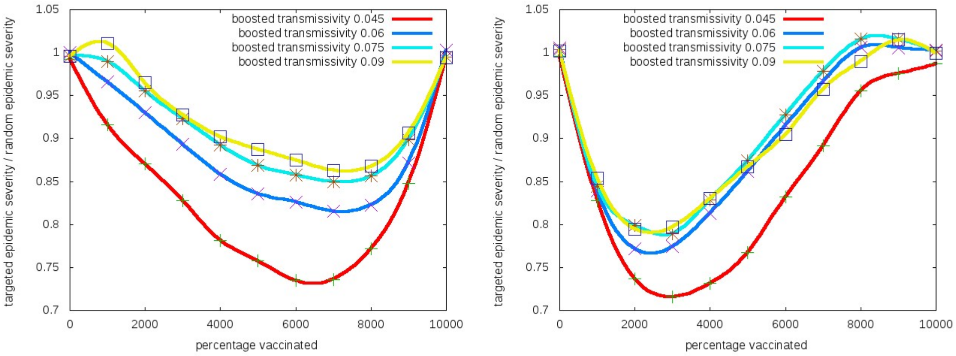


Figure 5.13: Epidemic severity comparison of random and targeted intervention strategies in one-sided (left) and two-sided (right) risk behavior models. x-axis is the percentage of nodes taking interventions, and y-axis is the ratio of the epidemic severity in targeted intervention strategy and the epidemic severity in random intervention strategy. $p = 0.03$, and $p_s = 0.35$.

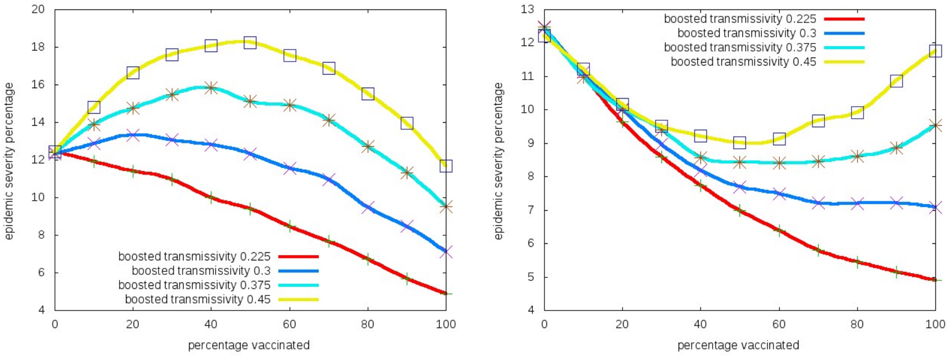


Figure 5.14: Epidemic severity with different boosted transmissivities in one-sided (left) and two-sided (right) risk behavior models. x-axis is the percentage of nodes taking interventions, and y-axis is the expected percentage of nodes getting infected. $p = 0.15$ and $p_s = 0.35$.

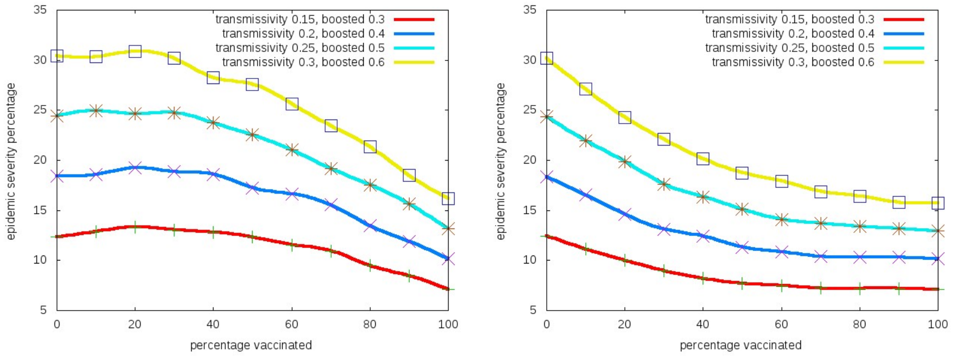


Figure 5.15: Epidemic severity with different transmissivities in one-sided (left) and two-sided (right) risk behavior models. x-axis is the percentage of nodes taking interventions, and y-axis is the expected percentage of nodes getting infected. $p_s = 0.35$, and $p_m = 2p$.

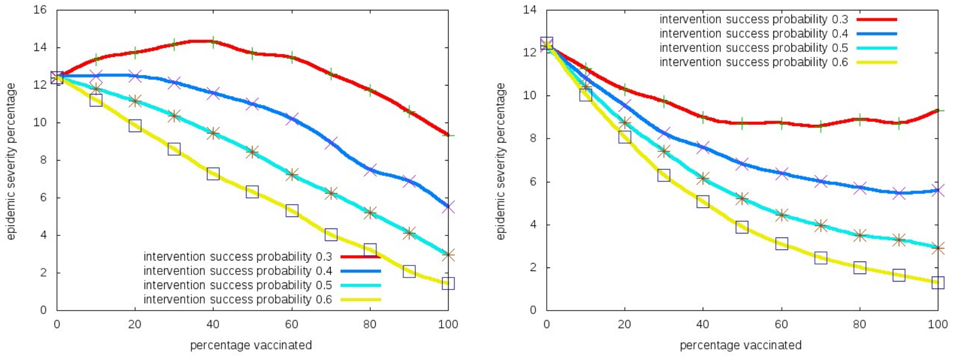


Figure 5.16: Epidemic severity with different intervention success probabilities in one-sided (left) and two-sided (right) risk behavior models. x-axis is the percentage of nodes taking interventions, and y-axis is the expected percentage of nodes getting infected. $p = 0.15$, and $p_m = 0.3$.

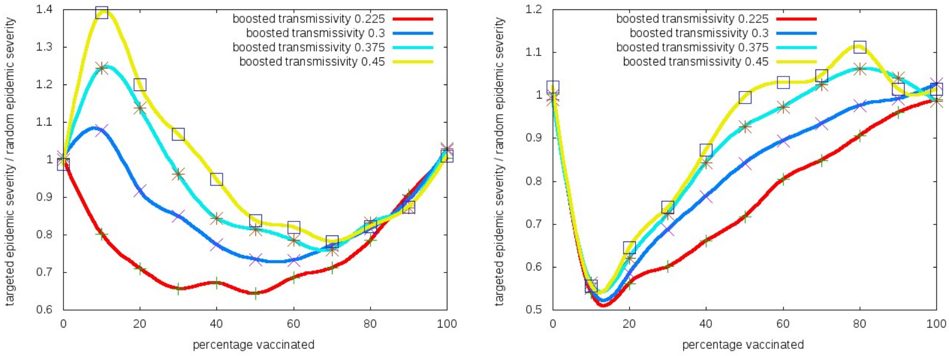


Figure 5.17: Epidemic severity comparison of random and targeted intervention strategies in one-sided (left) and two-sided (right) risk behavior models. x-axis is the percentage of nodes taking interventions, and y-axis is the ratio of the epidemic severity in targeted intervention strategy and the epidemic severity in random intervention strategy. $p = 0.15$, and $p_s = 0.35$.

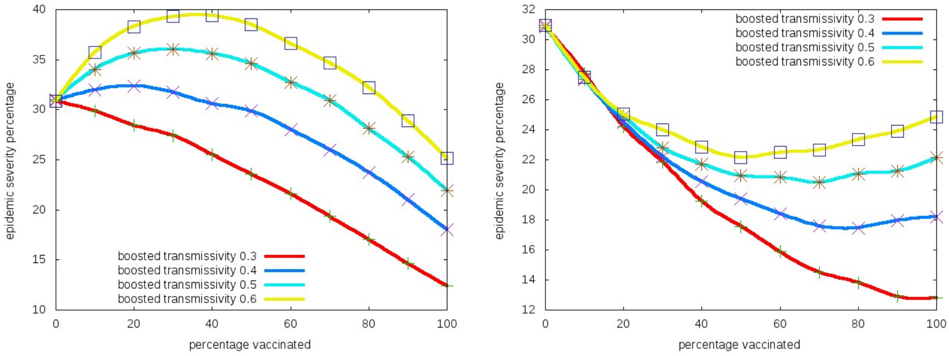


Figure 5.18: Epidemic severity with different boosted transmissivities in one-sided (left) and two-sided (right) risk behavior models. x-axis is the percentage of nodes taking interventions, and y-axis is the expected percentage of nodes getting infected. $p = 0.2$ and $p_s = 0.35$.

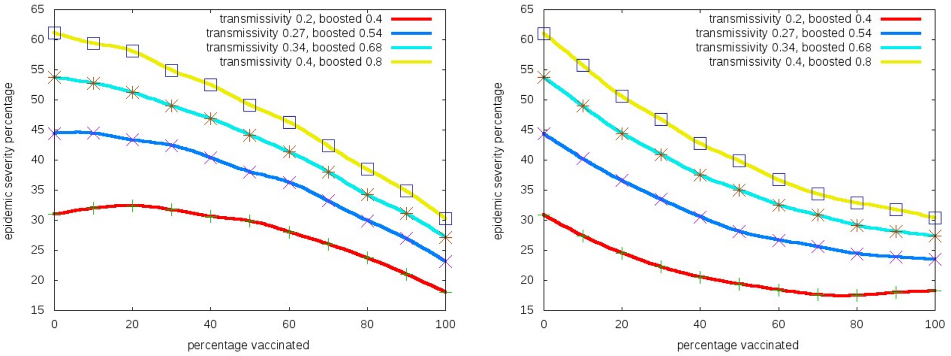


Figure 5.19: Epidemic severity with different transmissivities in one-sided (left) and two-sided (right) risk behavior models. x-axis is the percentage of nodes taking interventions, and y-axis is the expected percentage of nodes getting infected. $p_s = 0.35$, and $p_m = 2p$.

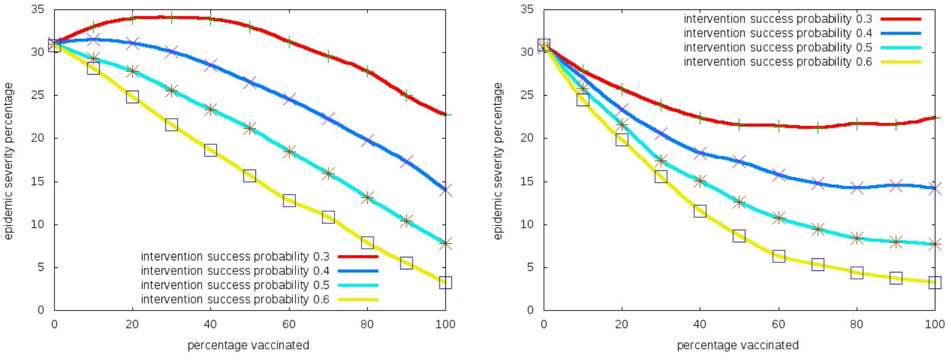


Figure 5.20: Epidemic severity with different intervention success probabilities in one-sided (left) and two-sided (right) risk behavior models. x-axis is the percentage of nodes taking interventions, and y-axis is the expected percentage of nodes getting infected. $p = 0.2$, and $p_m = 0.4$.

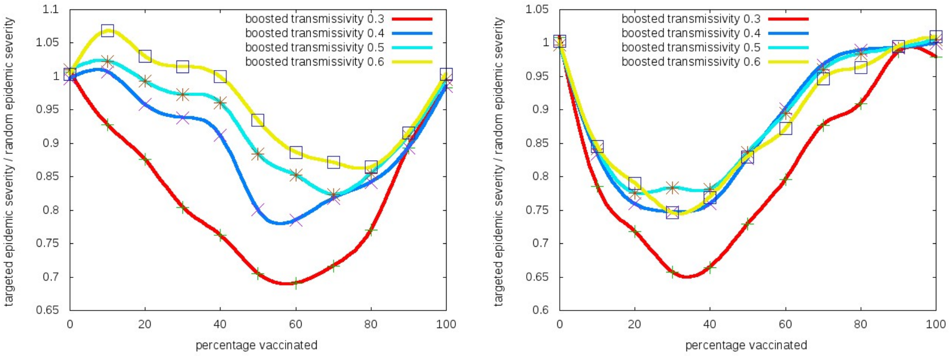


Figure 5.21: Epidemic severity comparison of random and targeted intervention strategies in one-sided (left) and two-sided (right) risk behavior models. x-axis is the percentage of nodes taking interventions, and y-axis is the ratio of the epidemic severity in targeted intervention strategy and the epidemic severity in random intervention strategy. $p = 0.2$, and $p_s = 0.35$.

be revisited in the context of new anti-retroviral treatments being considered for HIV [61].

Chapter 6

Conclusion

In this dissertation, we have developed and analyzed positive diffusion processes under both organic dynamics and adversarial dynamics. We have also designed and analyzed intervention strategies to control negative diffusion. We now conclude by summarizing our results and by presenting a few directions for future research.

6.1 Enable positive diffusion

In the first half of the dissertation, we considered the question of how to design efficient algorithms to enable positive diffusion.

- We first looked at diffusion under organic dynamics, where the communication network is altered by diffusion itself. We proposed two natural and simple processes, triangulation and two-hop walk. These two processes are good algorithms to solve resource discovery, group member discovery, and lots of other problems. We have shown in undirected graphs both processes complete in $O(n \log^2 n)$ rounds, and proved an $\Omega(n \log n)$ lower bound. This is an almost tight bound with a logarithmic gap. We also proved that in directed graphs, two-hop walk process completes in $O(n^2 \log n)$ rounds, and gave a matching lower bound $\Omega(n^2 \log n)$ in weakly connected directed graphs, and lower bound $\Omega(n^2)$ in strongly connected directed graphs.

- We then looked at diffusion under adversarial dynamics, more specifically, k-gossip problem. There we adopted the online adversary model proposed in [89], where Kuhn et al showed an $O(kn)$ upper bound and an $\Omega(n \log k)$ lower bound.

We improve their lower bound to $\Omega(kn/\log n)$, which is an almost tight bound. This suggests that under the adversarial model defined in [89], we cannot have efficient token-forwarding based algorithms. Thus, we designed an $O(n\sqrt{k \log n})$ algorithm and an bicriteria $(O(n^{\epsilon}), \log n)$ approximation algorithm under offline adversarial model, where the adversary has to give the whole sequence of communication graphs up in front.

6.2 Control negative diffusion

In the second half of this dissertation, we switch gear to ask the question of how to control negative diffusion.

- We first designed an $O(\log n)$ approximation algorithm for the optimal centralized intervention strategy, where the algorithm picks a set of nodes in the network to apply interventions in order to minimize the cost of negative diffusion.
- We then looked at the setting where nodes in the network make their own decision if they want to secure themselves. We used game theory to show the existence and non-existence of Nash equilibrium. We also compare the cost of decentralized solution with the optimal centralized solution, in other word, price of anarchy.
- Last, we analyze negative diffusion in the presence of risk behavior changes. We have observed and analyzed two counter intuitive phenomena: 1) less interventions can be more effective, and 2) targeted intervention strategy can be worse than random intervention strategy.

References

[1] Cdc flu vaccine effectiveness http://www.cdc.gov/flu/professionals/vaccination/effectivenessqa.h 13, 97

[2] Consumers dont relate bot infections to risky behavior as millions continue to click on spam http://www.maawg.org/consumers-donavior-millions-continue-click-spam. 14

[3] Enron email network http://snap.stanford.edu/data/email-enron.html. 95

[4] Gnutella peer-to-peer network http://snap.stanford.edu/data/p2p-gnutella04.html. 95

[5] I. Abraham and D. Dolev. Asynchronous resource discovery. *Computer Networks*, 50:1616–1629, July 2006. 3, 6

[6] M. Adler, E. Halperin, R. M. Karp, and V. V. Vazirani. A stochastic process on the hypercube with applications to peer-to-peer networks. In *STOC*, pages 575–584, 2003. 7

[7] Y. Afek, B. Awerbuch, and E. Gafni. Applying static network protocols to dynamic networks. In *FOCS*, pages 358–370, 1987. 40

[8] Y. Afek, E. Gafni, and A. Rosen. The slide mechanism with applications in dynamic networks. In *ACM PODC*, pages 35–46, 1992. 40

[9] A. Agarwal and M. Charikar. On the advantage of network coding for improving network throughput. In *Information Theory Workshop*, 2004. 41

[10] R. Ahlswede, N. Cai, S. Li, and R. Yeung. Network information flow. *Transactions on Information Theory*, 46(4):1204–1216, 2000. 41

[11] N. Alon. Problems and results in extremal combinatorics – ii. *DISCRETE MATHEMATICS*, 2003. 7

[12] N. Alon, A. Bar-Noy, N. Linial, and D. Peleg. A lower bound for radio broadcast. *Journal of Computer and System Sciences*, 43:290–298, 1991. 39

[13] K. Andersson, D. Owens, E. Vardas, G. Gray, J. McIntyre, and A. Paltiel. Predicting the impact of a partially effective HIV vaccine and subsequent risk behavior change on the heterosexual HIV epidemic in low- and middle-income countries: A South African example. *J Acquir Immune Defic Syndr.*, 46(1):78–90, 2007. 13

[14] S. Arora, S. Rao, and U. Vazirani. Expander flows, geometric embeddings and graph partitioning. In *Proceedings of ACM STOC*, pages 222–231, 2004. 59

[15] J. Aspnes, K. L. Chang, and A. Yampolskiy. Inoculation strategies for victims of viruses and the sum-of-squares partition problem. *J. Comput. Syst. Sci.*, 72(6):1077–1093, 2006. 10, 11, 57, 58, 60, 61, 64, 65, 70

[16] J. Aspnes, N. Rustagi, and J. Saia. Worm versus alert: Who wins in a battle for control of a large-scale network? In *Principles of Distributed Systems: 11th International Conference, OPODIS 2007*, volume 4878 of *Lecture Notes in Computer Science*, pages 443–456. Springer, Dec 2007. 10, 58

[17] H. Attiya and J. Welch. *Distributed Computing: Fundamentals, Simulations and Advanced Topics (2nd edition).* John Wiley Interscience, March 2004. 39, 40

[18] J. Augustine, G. Pandurangan, P. Robinson, and E. Upfal. Towards robust and efficient computation in dynamic peer-to-peer networks. In *ACM-SIAM SODA*, 2012. 40

[19] C. Avin, M. Borokhovich, K. Censor-Hillel, and Z. Lotker. Order optimal information spreading using algebraic gossip. In *ACM PODC*, pages 363–372, 2011. 41

[20] C. Avin, M. Koucký, and Z. Lotker. How to explore a fast-changing world (cover time of a simple random walk on evolving graphs). In *ICALP*, pages 121–132, 2008. 41

[21] B. Awerbuch, P. Berenbrink, A. Brinkmann, and C. Scheideler. Simple routing strategies for adversarial systems. In *IEEE FOCS*, pages 158–167, 2001. 40

[22] B. Awerbuch and T. Leighton. Improved approximation algorithms for the multi-commodity flow problem and local competitive routing in dynamic networks. In *ACM STOC*, pages 487–496, May 1994. 40

[23] B. Awerbuch, B. Patt-Shamir, D. Peleg, and M. Saks. Adapting to asynchronous dynamic networks. In *STOC*, pages 557–570, 1992. 40

[24] Baruch Awerbuch, Andr Brinkmann, and Christian Scheideler. Anycasting in adversarial systems: Routing and admission control. In *ICALP*, pages 1153–1168, 2003. 40

[25] A. Bar-Noy, S. Guha, J. Naor, and B. Schieber. Message multicasting in heterogeneous networks. *SIAM J. Comput.*, pages 347–358, 2000. 39

[26] R. Bar-Yehuda, O. Goldreich, and A. Itai. On the time-complexity of broadcast in radio networks: an exponential gap between determinism and randomization. In *ACM PODC*, pages 98–108, 1987. 39

[27] A. L. Barabási and R. Albert. Emergence of scaling in random networks. In *Science*, volume 286, 1999. 76

[28] A.L. Barabási and R. Albert. Emergence of scaling in random networks. *Science*, 1999. 84, 95

[29] C. Barrett, D. Beckman, M. Khan, V.S. Anil Kumar, M. Marathe, P. Stretz, T. Dutta, and B. Lewis. Generation and analysis of large synthetic social contact networks. In *Winter Simulation Conference*. 95

[30] M. Barthelemy, A. Barrat, R. Pastor-Satorras, and A.Vespignani. Dynamical patterns of epidemic outbreaks in complex heterogeneous networks. *Journal of Theoretical Biology*, pages 275–288, 2005. 59

[31] C. Bauch. Imitation dynamics predict vaccinating behaviour. In *Proceeding of The Royal Society*, 2005. 59

[32] C. Bauch and D. Earn. Vaccination and the theory of games. *PNAS*, 101:13391–13394, September 2004. 59

[33] C. Bauch, A. Galvani, and D. Earn. Group interest versus self-interest in smallpox vaccination policy. *PNAS*, 2003. 59

[34] P. Berenbrink, J. Czyzowicz, R. Elsässer, and L. Gasieniec. Efficient information exchange in the random phone-call model. In *ICALP*, pages 127–138, 2010. 41

[35] N. Berger, C. Borgs, J. T. Chayes, and A. Saberi. On the spread of viruses on the internet. In *Proceedings of SODA*, 2005. 10, 58, 94

[36] D. Bezemer, F. Wolf, M.C. Boerlijst, A. Sighem, T.D. Hollingsworth, M. Prins, R.B. Geskus, L. Gras, R. Goutinho, and C. Fraser. A resurgent HIV-1 epidemic among men who have sex with men in the era of potent antiretroviral therapy. *AIDS*, 2008. 13

[37] S.M. Blower and A.R. McLean. Prophylactic vaccination, risk behavior change, and the probability of eradicating HIV in San Francisco. *Science*, 1994. 13, 15

[38] B. Bollobás. *Percolation Theory.* Cambridge University Press, 2006. 89

[39] B. Bollobás, S. Janson, and O. Riordan. The phase transition in inhomogeneous random graphs. *Random Structures Algorithms*, 2007. 88

[40] S. Bornholdt and H. Schuster (Editors). *Handbook of Graphs and Networks.* Wiley-VCH, 2003. 2

[41] O. V. Borodin, A. V. Kostochka, and B. Toft. Variable degeneracy: Extensions of Brooks' and Gallai's theorems. *Discrete Mathematics*, 214(1-3):101–112, 2000. 61, 63

[42] M. Borokhovich, C. Avin, and Z. Lotker. Tight bounds for algebraic gossip on graphs. In *IEEE ISIT*, 2010. 41

[43] S. Boyd, A. Ghosh, B. Prabhakar, and D. Shah. Randomized gossip algorithms. *IEEE Trans. on Infor. Theory*, 52(6):2508–2530, 2006. 3, 41

[44] N. Brewer, C. Cuite, J. Herrington, and N. Weinstein. Risk compensation and vaccination: can getting vaccinated cause people to engage in risky behaviors? *Ann. Behav. Med.*, 34(1):95–9, 2007. 13

[45] S. Chakrabarti, A. Frieze, and J. Vera. The influence of search engines on preferential attachment. In *SODA*, 2005. 3

[46] M. Charikar, C. Chekuri, T. Cheung, Z. Dai, A. Goel, and S. Guha. Approximation algorithms for directed steiner problems. *Journal of Algorithms*, 1998. 41

[47] J. Chen and G. Pandurangan. Optimal gossip-based aggregate computation. In *SPAA*, pages 124–133, 2010. 3, 41

[48] J. Cheriyan and M. Salavatipour. Hardness and approximation results for packing steiner trees. *Algorithmica*, 2006. 41, 53, 54, 55

[49] G. Chowell, H. Nishiura, and L. Bettencourt. Comparative estimation of the reproduction number for pandemic influenza from daily case notification data. *The Royal Society*, 2006. 97

[50] Andrea E. F. Clementi, Angelo Monti, and Riccardo Silvestri. Distributed multi-broadcast in unknown radio networks. In *PODC*, pages 255–264, 2001. 39

[51] C. Cooper and A. Frieze. Crawling on web graphs. In *STOC*, 2002. 3

[52] R. Crosby and D. Holtgrave. Will sexual risk behaviour increase after being vaccinated for AIDS? *Int J STD AIDS*, 17(3):180–4, 2006. 13

[53] S. Deb, M. Médard, and C. Choute. Algebraic gossip: a network coding approach to optimal multiple rumor mongering. *IEEE/ACM Trans. Netw.*, 14:2486–2507, June 2006. 41

[54] A. Demers, D. Greene, C. Hauser, W. Irish, J. Larson, S. Shenker, H. Sturgis, D. Swinehart, and D. Terry. Epidemic algorithms for replicated database maintenance. In *PODC*, pages 1–12, 1987. 3, 41

[55] Z. Dezso and A.L. Barabási. Halting viruses in scale-free networks. *Physical Review E*, 2002. 14, 94

[56] N. B. Dimitrov and C. Greg Plaxton. Optimal cover time for a graph-based coupon collector process. In *ICALP*, pages 702–716, 2005. 7

[57] S. Dolev. *Self-stabilization.* MIT Press, Cambridge, MA, USA, 2000. 40

[58] H. Ebel, L. Mielsch, and S. Bornholdt. Scale-free topology of e-mail networks. *Phys. Rev. E*, 66, 2002. 76

[59] S. Eidenbenz, A. Kumar, and S. Zust. Equilibria in topology control games for ad hoc networks. *MONET*, 11(2):143–159, 2006. 58

[60] P. Erdös and A. Rényi. On the evolution of random graphs. *Publ. Math.*, 1960. 90, 91, 95

[61] Q.A. Karim et al. Effectiveness and safety of tenofovir gel, an antiretroviral microbicide, for the prevention of HIV infection in women. *Science*, 329 (5996):1168–1174, 2010. 13, 97, 105

[62] M. Faloutsos, P. Faloutsos, and C. Faloutsos. On power-law relationships of the internet topology. In *Proceedings of SIGCOMM*, pages 251–262, 1999. 76

[63] U. Feige. Vertex cover is hardest to approximate on regular graphs. Technical Report MCS03-15, The Weizmann Institute of Science, 2003. 72

[64] A. Forster, J. Wardle, J. Stephenson, and J. Waller. Passport to promiscuity or lifesaver: press coverage of hpv vaccination and risky sexual behavior. *J. Health Commun.*, 15(2):205–17, 2010. 13

[65] C. Fraser, C. Donnelly, S. Cauchemez, and et al. Pandemic potential of a strain of influenza a (h1n1): Early findings. *Science*, 2009. 97

[66] E. Gafni and B. Bertsekas. Distributed algorithms for generating loop-free routes in networks with frequently changing topology. *IEEE Trans. Comm.*, 29(1), 1981. 40

[67] A. Ganesh, L. Massoulie, and D. Towsley. The effect of network topology on the spread of epidemics. In *Proceeding of INFOCOM 2005*, 2005. 10, 14, 58

[68] N. Garg, V. Vazirani, and M Yannakakis. Approximate max-flow min-(multi)cut theorems and their applications. *SIAM Journal on Computing*, 25:698–707, 1993. 75

[69] N.C. Grassly and C. Fraser. Mathematical models of infectious disease transmission. *Nature*, 2008. 14

[70] R. Gray, X. Li, M. Wawer, S. Gange, D. Serwadda, N. Sewankambo, R. Moore, F. Wabwire-Mangen, T. Lutalo, and T. Quinn. Stochastic simulation of the impact of antiretroviral therapy and HIV vaccines on HIV transmission; Rakai, Uganda. *AIDS*, 17(13):1941–51, 2003. 13, 97

[71] J. Grossklags, N. Christin, and J. Chuang. Secure or insure? a game-theoretic analysis of information security games. In *World Wide Web Conference (WWW)*, 2008. 10, 58

[72] B. Haeupler. Analyzing network coding gossip made easy. In *ACM STOC*, pages 293–302, 2011. 9, 41

[73] B. Haeupler and D. Karger. Faster information dissemination in dynamic networks via network coding. In *ACM PODC*, pages 381–390, 2011. 9, 41, 56

[74] M. Halloran, I. Longini, M. Haber, C. Struchiner, and R. Brunet. Exposure efficacy and change in contact rates in evaluating prophylactic HIV vaccines in the field. *Stat Med.*, 13(4):357–77, 1994. 13

[75] M. Harchol-balter, T. Leighton, and D. Lewin. Resource discovery in distributed networks. In *Symposium on Principles of Distributed Computing*, pages 229–237, 1999. 3, 6

[76] M. Jackson and L. Yariv. Diffusion, strategic interaction, and social structure. In *The Handbook of Social Economics*, 2010. 14

[77] L. Jia, R. Rajaraman, and C. Scheideler. On local algorithms for topology control and routing in ad hoc networks. In *ACM SPAA*, pages 220–229, June 2003. 40

[78] R. Kannan, S. Vempala, and A. Vetta. On clusterings: Good, bad and spectral. *J. ACM*, 51(3):497–515, 2004. 81

[79] E.H. Kaplan, D.L. Craft, and L.M. Wein. Emergency response to a smallpox attack: The case for mass vaccination. *PNAS*, 2002. 14

[80] R. Karp, C. Schindelhauer, S. Shenker, and B. Vöcking. Randomized rumor spreading. In *FOCS*, pages 565–574, 2000. 3, 41

[81] M. Kearns and L. Ortiz. Algorithms for interdependent security games. In *Advances in Neural Information Processing Systems, MIT Press*, 2004. 11, 59

[82] D. Kempe, A. Dobra, and J. Gehrke. Gossip-based computation of aggregate information. In *FOCS*, pages 482–491, 2003. 3, 41

[83] D. Kempe and J. Kleinberg. Protocols and impossibility results for gossip-based communication mechanisms. In *Proceedings of FOCS*, 2002. 3

[84] D. Kempe, J. Kleinberg, and A. Demers. Spatial gossip and resource location protocols. In *STOC*, 2001. 3

[85] D. A. Kessler. Epidemic size in the SIS model of endemic infections. *J. Appl. Probab.*, 43:757–778, 2008. 59

[86] S. Klein. Parasite manipulation of the proximate mechanisms that mediate social behavior in vertebrates. *Physiol Behav.*, 79:441–9, 2003. 13

[87] B. Klimt and Y. Yang. Introducing the enron corpus. *CEAS*, 2004. 95

[88] E. Koutsoupias and C. H. Papadimitriou. Worst-case equilibria. In *Proceedings of STACS*, 1999. 61

[89] F. Kuhn, N. Lynch, and R. Oshman. Distributed computation in dynamic networks. In *ACM STOC*, pages 513–522, 2010. 8, 9, 37, 38, 39, 40, 41, 107, 108

[90] F. Kuhn and R. Oshman. Dynamic networks: Models and algorithms. *SIGACT News*, 42(1):82–96, 2011. 40

[91] S. Kutten, D. Peleg, and U. Vishkin. Deterministic resource discovery in distributed networks. In *SPAA*, 2001. 6

[92] C. Law and K. Siu. An $o(\log n)$ randomized resource discovery algorithm. In *14th International Symposium on Distributed Computing (Brief Announcement), Technical Report, Technical University of Madrid*, pages 5–8, 2000. 3, 6

[93] T. Leighton. *Introduction to Parallel Algorithms and Architectures: Arrays, Trees, and Hypercubes.* Morgan-Kaufmann, San Mateo, CA, 1991. 39

[94] M. Lelarge and J. Bolot. Economic incentives to increase security in the internet: The case for insurance. In *IEEE Infocom*, 2009. 10, 58

[95] A. L. Lloyd and R. M. May. Epidemiology. how viruses spread among computers and people. *Science*, page 1316, 2001. 14

[96] N. Lynch. *Distributed Algorithms.* Morgan Kaufmann, San Francisco, CA, 1996. 39, 40

[97] C.F. Manski. Vaccination with partial knowledge of external effectiveness. *PNAS*, 2010. 14

[98] A.R. McLean and S.M. Blower. Imperfect vaccines and herd immunity to HIV. *Proceedings of the Royal Society of London*, 1993. 13, 14, 15

[99] J. Medlock and A.P. Galvani. Optimizing influenza vaccine distribution. *Science*, 2009. 14

[100] S. Mei, R. Quax, D. van de Vijver, Y. Zhu, and P. Sloot. Increasing risk behaviour can outweigh the benefits of antiretroviral drug treatment on the HIV incidence among men-having-sex-with-men in Amsterdam. *BMC Infectious Diseases 2011*, 11:118, 2011. 13

[101] L.A. Meyers. Contact network epidemiology: Bond percolation applied to infectious disease prediction and control. *American Mathematical Society*, 2006. 14

[102] L.A. Meyers, B. Pourbohloul, M.E.J. Newman, D.M. Skowronski, and R.C. Brunham. Network theory and SARS: Predicting outbreak diversity. *Theoretical Biology*, 2005. 14

[103] J. A. Mirrlees. The theory of moral hazard and unobservable behaviour: Part i. *Review of Economic Studies*, 66:321, 1999. 13

[104] M. Mitzenmacher and E. Upfal. *Probability and Computing: Randomized Algorithms and Probabilistic Analysis.* Cambridge University Press, 2004. 19

[105] J. Moore. *Parasites and the behavior of animals.* Oxford University Press, New York, 2002. 13

[106] D. Mosk-Aoyama and D. Shah. Computing separable functions via gossip. In *PODC*, pages 113–122, 2006. 3, 41

[107] A. Nahir and A. Orda. Topology design and control: A game-theoretic perspective. In *IEEE INFOCOM*, 2009. 58

[108] M. Newman. The structure and function of complex networks. *SIAM Review*, 45, 2003. 14, 85

[109] M. J. Newman, A. Barabasi, and D. J. Watts. *Structure and Dynamics of Networks.* Princeton University Press, 2006. 2

[110] M.E. Newman. Spread of epidemic disease on networks. *Phys. Rev. E*, 2002. 14

[111] N. Nisan, T. Roughgarden, E. Tardos, and V. Vazirani. *Algorithmic Game Theory.* Cambridge University Press, 2008. 58, 81

[112] J. Omic, A. Orda, and P. Mieghem. Protecting against network infections: A game theoretic perspective. In *IEEE Infocom*, 2009. 58, 59

[113] R. Pastor-Satorras and A. Vespignani. Epidemic spreading in scale-free networks. *Phys. Rev. Lett.*, 2000. 14

[114] J. Petrie, S. Ohmit, E. Johnson, R. Cross, and A. Monto. Efficacy studies of influenza vaccines: Effect of end points used and characteristics of vaccine failures. *Journal of Infectious Diseases*, 2011. 13

[115] C. Reiber, E. Shattuck, S. Fiore, P. Alperin, V. Davis, and J. Moore. Change in human social behavior in response to a common vaccine. *Ann. Epidemiol.*, 20(10):729–33, 2010. 13

[116] M. Ripeanu, I. Foster, and A. Iamnitchi. Mapping the gnutella network: Properties of large-scale peer-to-peer systems and implications for system design. *IEEE Internet Computing Journal*, 2002. 95

[117] T. Roughgarden and E. Tardos. How bad is selfish routing? *J. ACM*, page 236259, 2002. 58

[118] P. Sanders, S. Egner, and L. Tolhuizen. Polynomial time algorithms for network information flow. In *ACM SPAA*, pages 286–294, 2003. 41

[119] R. Smith and S. Blower. Could disease-modifying HIV vaccines cause population-level perversity? *Lancet Infect Dis.*, 4(10):636–9, 2004. 13

[120] B. Söderberg. A general formalism for inhomogeneous random graphs. *Phys. Rev. E*, 2002. 88, 90, 92

[121] A. Straten, C. Gómez, J. Saul, J. Quan, and N. Padian. Sexual risk behaviors among heterosexual HIV serodiscordant couples in the era of post-exposure prevention and viral suppressive therapy. *AIDS*, 14(4):F47–54, 2000. 13

[122] D. Topkis. Concurrent broadcast for information dissemination. *IEEE Trans. Softw. Eng.*, 11:1107–1112, October 1985. 39

[123] F. Vega-Redondo. *Complex Social Networks*. Cambridge University Press, 2007. 2

[124] J.X. Velasco-Hernandez, H.B. Gershengorn, and S.M. Blower. Could widespread use of combination antiretroviral therapy eradicate HIV epidemics? *Infectious Diseases*, 2002. 13

[125] E. Vynnycky, A. Trindall, and P. Mangtani. Estimates of the reproduction numbers of spanish influenza using morbidity data. *International Journal of Epidemiology*, 2007. 97

[126] Y. Wang, D. Chakrabarti, C. Wang, and C. Faloutsos. Epidemic spreading in real networks: An eigenvalue viewpoint. In *In SRDS*, pages 25–34, 2003. 10, 58

[127] Y. Wang, G. Xiao, J. Hu, T.H. Cheng, and L. Wang. Imperfect targeted immunization in scale-free networks. *Physica A*, 2009. 14, 94

[128] D.P. Wilson, P.M. Coplan, M.A. Wainberg, and S.M. Blower. The paradoxical effects of using antiretroviral-based microbicides to control HIV epidemics. *PNAS*, 2008. 13

[129] Y. Yang, J.D. Sugimoto, M.E. Halloran, N.E. Basta, D.L. Chao, L. Matrajt, G. Potter, E. Kenah, and I.M. Longini. The transmissibility and control of pandemic influenza A (H1N1) virus. *Science*, 2009. 14

[130] L. Zosin and S. Khuller. On directed Steiner trees. In *Proceedings of the 13th Annual ACM-SIAM Symposium on Discrete Algorithms*, January 2002. 41