Hiding the Rumor Source

Giulia Fanti, Peter Kairouz, *Member, IEEE,* Sewoong Oh, *Member, IEEE,* Kannan Ramchandran, *Fellow, IEEE*
and Pramod Viswanath, *Fellow, IEEE,*

***Abstract*—Anonymous social media platforms like Secret, Yik Yak, and Whisper have emerged as important tools for sharing ideas without the fear of judgment. Such anonymous platforms are also important in nations under authoritarian rule, where freedom of expression and the personal safety of message authors may depend on anonymity. Whether for fear of judgment or retribution, it is sometimes crucial to hide the identities of users who post sensitive messages. In this paper, we consider a global adversary who wishes to identify the author of a message; it observes either a snapshot of the spread of a message at a certain time, sampled timestamp metadata, or both. Recent advances in rumor source detection show that existing messaging protocols are vulnerable against such an adversary. We introduce a novel messaging protocol, which we call adaptive diffusion, and show that under the snapshot adversarial model, adaptive diffusion spreads content fast and achieves perfect obfuscation of the source when the underlying contact network is an infinite regular tree. That is, all users with the message are nearly equally likely to have been the origin of the message. When the contact network is an irregular tree, we characterize the probability of maximum likelihood detection by proving a concentration result over Galton-Watson trees. Experiments on a sampled Facebook network demonstrate that adaptive diffusion effectively hides the location of the source even when the graph is finite, irregular, and has cycles.**

***Index Terms*—privacy, diffusion, anonymous social media.**

I. Introduction

Microblogging platforms are central to the fabric of the present Internet; popular examples include Twitter and Facebook. In such platforms, users propagate short messages (texts, images, videos) over a contact graph, which represents a social network in most cases. Message forwarding often occurs through built-in mechanisms that rely on user input, such as clicking “like" or “share" on a particular post. Brevity of message, fluidity of user interface, and trusted party communication combine to make these microblogging platforms a major communication mode of modern times.

However, the popularity of microblogging services also makes them a prime target for invasive user monitoring by employers, service providers, or government agencies. This monitoring typically exploits *metadata*: non-content data that characterizes content, like timestamps. Metadata can often be as sensitive as data itself [1], [2]; this reality was publicized by Michael Hayden, former Director of the CIA, with his observation that “We kill people based on metadata" [3].

The alarming privacy implications of microblogging platforms have spurred the growth of *anonymous* microblogging platforms, like Whisper [4], Yik Yak [5], and the now-defunct Secret [6]. These platforms enable users to share messages with their friends without revealing authorship metadata. Although anonymous messaging platforms are often used for social privacy (i.e., hiding authorship from from one’s peers), they can also be useful in states with authoritarian governments, where the right to free expression, and sometimes the personal safety of message authors, depend on anonymity. In such scenarios, it is often crucial to hide the identity of the user who initially posted the message.

Whether for fear of judgment or personal endangerment, protecting of the identity of content sources is becoming increasingly important. However, existing anonymous messaging services store both messages and authorship information on centralized servers, which makes them vulnerable to government subpoenas, hacking, or direct company access. An alternative solution would be to store this information in a distributed fashion; each node would know only its own friends, and message authorship information would never be transmitted to any party. Distributed systems are more robust to monitoring due to lack of central points of failure. However, even under distributed architectures, simple anonymous messaging protocols (such as those used by commercial anonymous microblogging apps) are still vulnerable against an adversary with side information, as proved in recent advances in rumor source detection [7], [8]. In this work, we study a basic building block of the messaging protocol that would underpin truly anonymous microblogging platform – *how to anonymously broadcast a single message on a contact network*, even in the face of a strong deanonymizing adversary with access to metadata. Specifically, we focus on anonymous microblogging built atop an underlying *social network*, such as a network of phone contacts or Facebook friends.

Adversaries. We consider three adversarial models, which capture different approaches to collecting metadata. In each case, the underlying contact network is modeled as a graph that is known to the adversary. Beyond that, the adversary could proceed in a few different ways.

The adversary might use side channels to infer whether a node is infected, i.e., whether it received the message. If an adversary collects only infection metadata for all network users, we call it a *snapshot* adversary. This could represent a state-level adversary that attends a Twitter-organized protest; it implicitly learns who received the protest advertisement by observing which individuals are physically present, but not the

G. Fanti, P. Kairouz, S. Oh, and P. Viswanath are with the University of Illinois at Urbana-Champaign, Urbana, IL 61801 USA e-mail: {fanti,kairouz2,swoh,pramodv}@illinois.edu .

K. Ramchandran is with the Department of Electrical and Computer Engineering, University of California, Berkeley, CA, 30332 USA e-mail: kannanr@eecs.berkeley.edu .

This work was presented in part at ACM Sigmetrics 2015 and 2016.

associated metadata. The snapshot adversary is well-studied in the literature, primarily in the related problem of *source identification* [7], [9], [10], [11], [12]. We focus primarily on the snapshot adversarial model in this paper.

Alternatively, the adversary might explicitly corrupt some fraction of nodes by bribery or coercion; these corrupted *spy nodes* could pass along metadata like message timestamps and relay IDs. If an adversary only collects information from spies, we call it a *spy-based* adversary. A spy-based adversary could represent a government agency participating in social media to study users, for instance. The adversary's reach may be limited by factors like account creation, contact network structure [13], or the cost of corrupting participants. This adversarial model is discussed in detail in [14], but we include the relevant theoretical results in this paper for the sake of completeness.

Finally, an adversary could combine the spy-based and snapshot adversarial models by using both forms of metadata. If an adversary uses spies and a snapshot, we call it a *spy+snapshot* adversary. This adversarial model allows us to study the capabilities of both snapshot and spy metadata types, combining the results on snapshot adversary capabilities derived here with those of spy adversary capabilities derived in [14].

Spreading models. In social networks, messages are typically propagated based on users' approval, which is expressed via liking, sharing or retweeting. This mechanism, which enables social filtering and reduces spam, has inherent random delays associated with each user's time of impression and decision to “like" the message (or not). Standard models of rumor spreading in networks explicitly model such random delays via a *diffusion* process: messages are spread independently over different edges with a fixed probability of spreading (discrete time model) or an exponential spreading time (continuous time model). The designer can partially control the spreading rate by introducing artificial delays on top of the usual random delays due to users' approval of the messages.

We model this physical setup as a discrete-time system. At time $t = 0$, a single user $v^* \in V$ starts to spread a message over the contact network $G = (V, E)$ where users and contacts are represented by nodes and edges, respectively. Upon receiving the message, nodes approve it immediately. The assumption that all nodes are willing to approve and pass the message is common in rumor spreading analysis [7], [15], [16]. However, by assuming message approval is immediate, we abstract away the natural random delays typically modeled by diffusion. At the following timestep, the protocol decides which neighbors will receive the message, and how much propagation delay to introduce. Given this control, the system designer wishes to design a spreading protocol that makes message source inference difficult.

Specifically, after T timesteps, let $V_T \subseteq V$, G_T, and $N_T \triangleq |V_T|$ denote the set of infected nodes, the subgraph of G containing only V_T, and the number of infected nodes, respectively. At a given time T, the adversary uses all available metadata to estimate the source. We assume no prior knowledge of the source, so the adversary computes a maximum-likelihood (ML) estimate of the source $\hat{v}_{\text{ML}}$. We desire a spreading protocol that minimizes the probability of detection $P_D = \mathbb{P}(\hat{v}_{\text{ML}} = v^*)$.

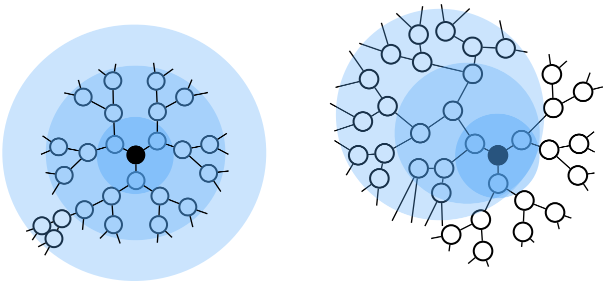


Fig. 1: Illustration of a spread of infection when spreading immediately (left) and under adaptive diffusion (right).

Current state-of-the-art: Diffusion is commonly used to model epidemic propagation over contact networks. While simplistic (it ignores factors like individual user preferences), diffusion is a commonly-studied and useful model due to its simplicity and first-order approximation of actual propagation dynamics. Critically, it captures the *symmetric* spreading used by most social media platforms.

However, diffusion has been shown to exhibit poor anonymity properties; under the adversarial models we consider, the source can be identified reliably [7], [8]. We therefore seek a different spreading model with strong anonymity guarantees. We wish to achieve the following performance metrics:

(a) We say a protocol has an *order-optimal rate of spread* if the expected time for the message to reach n nodes scales linearly compared to the time required by the fastest spreading protocol.

(b) We say a protocol achieves a *perfect obfuscation* if the probability of source detection for the maximum likelihood estimator is order-optimal. The definition of optimality differs for different adversarial models, so we define this metric separately for each adversarial model.

Contributions. We introduce *adaptive diffusion*, a novel messaging protocol with provable author anonymity guarantees against all of the discussed adversarial models. Whereas diffusion spreads the message symmetrically in all directions, adaptive diffusion breaks that symmetry (Figure 1). This has different implications for different adversarial models, but it consistently yields stronger anonymity guarantees than diffusion. Adaptive diffusion is also inherently distributed and spreads messages fast, i.e., the time it takes adaptive diffusion to reach n users is at most twice the time it takes the fastest spreading scheme which immediately passes the message to all its neighbors.

We prove that over d-regular trees, adaptive diffusion provides perfect obfuscation of the source under the snapshot adversarial model. That is, the likelihood of an infected user being the source of the infection is equal among all infected users. We derive exact expressions for the probability of detection, and show that this expression is optimal for the snapshot adversary by providing a matching fundamental lower bound.

In practice, the contact networks are not regular infinite trees. For a general class of graphs which can be finite, irregular and have cycles, we provide results of numerical experiments on real-world social networks and synthetic networks showing that the protocol hides the source at nearly the best possible level of obfuscation under the snapshot adversarial model. The same is true for spy-based adversaries; such simulation results for that adversarial model are discussed in [14]. Further, for a specific family of random *irregular* infinite trees, known as Galton-Watson trees, we characterize the probability of detection under adaptive diffusion and a snapshot adversary. In the process, we prove a strong concentration for the extreme paths in the Galton-Watson tree that consists of nodes with large degrees, which might be of independent interest.

Finally, we characterize the probability of maximum likelihood detection of adaptive diffusion in various edge cases, such as when the adversary can take *multiple snapshots*, and when the underlying graph contains regular cycles, as in an infinite lattice graph.

Related work. Anonymous communication has been a popular research topic for decades. For instance, anonymous point-to-point communication allows a sender to communicate with a receiver without the receiver learning the sender's identity. A great deal of work has emerged in this area, including Tor [17], Freenet [18], Free Haven [19], and Tarzan [20]. In contrast to this body of work, we address the problem of anonymously *broadcasting* a message over an underlying contact network (e.g., a network of Facebook friendships or phone contacts).

Anonymous broadcast communication has been most studied in context of the dining cryptographers' (DC) problem. We diverge from the literature on this topic [21], [22], [23], [24], [25] in approach and formulation. We consider statistical spreading models rather than cryptographic encodings, accommodate computationally unbounded adversaries, and consider domain-specific contact networks rather than a fully connected communication network.

Recently, Riposte addressed a similar problem of anonymously writing to a public message board [26]. It uses techniques from private information retrieval to store multiple, corrupted copies of messages on distributed servers. This corruption is designed so that no subset of colluding servers can determine the author. However, Riposte places no restrictions on communication with the servers, thereby facilitating spam. Differences in the communication model and adversarial model prevent Riposte from effectively solving our problem of interest.

Within the realm of statistical message spreading models, the problem of detecting the origin of an epidemic or the source of a rumor has been studied under the *diffusion* model. Recent advances in [7], [27], [9], [10], [11], [12], [16], [28] show that it is possible to identify the source within a few hops with high probability. Consider an adversary who has access to the underlying *contact network* of friendship links and the snapshot of infected nodes at a certain time. The problem of locating a rumor source, first posed in [7], naturally corresponds to graph-centrality-based inference algorithms: for a continuous time model, [7], [27] used the rumor centrality measure to correctly identify the source after time T (with probability converging to a positive number for large d-regular and random trees, and with probability proportional to $1/\sqrt{T}$ for lines). The probability of identifying the source increases even further when multiple infections from the same source are observed [9]. With multiple sources of infections, spectral methods have been proposed for estimating the number of sources and the set of source nodes in [10], [11]. When infected nodes are allowed to recover as in the susceptible-infected-recovered (SIR) model, Jordan centrality was proposed in [12], [16] to estimate the source. In [16], it is shown that the Jordan center is still within a bounded hop distance from the true source with high probability, independent of the number of infected nodes. When the adversary collects timestamps (and other metadata) from spy nodes, standard diffusion reveals the location of the source [8], [16], [29]. However, ML estimation is known to be NP-hard [30], and analyzing the probability of detection is also challenging. Finally, there has been work on related problems of characterizing the generative spreading model for an observed infection [31], [32], [33] as well the underlying network [34], [35].

In summary, under natural, diffusion-based message spreading—as seen in almost every content-sharing platform today—an adversary with some side information can identify the rumor source with high confidence. We overcome this vulnerability by asking the reverse question: can we design messaging protocols that spread fast while protecting the anonymity of the source? Related challenges include (a) identifying the best algorithm that the adversary might use to infer the location of the source; (b) providing analytical guarantees for the proposed spreading model; and (c) identifying the fundamental limit on what any spreading model can achieve. We address all of these challenges for snapshot adversarial model (Section IV), spy-based adversarial model (Section V), and finally the spy+snapshot model (Section VI). In this paper, our primarily focus is on the snapshot adversarial model; the spy-based and spy+snapshot adversaries are discussed in detail in [14].

Our work fits into a larger ecosystem that enables anonymous messaging; we implicitly assume that the ecosystem is healthy. For instance, we assume that nodes communicate securely in a distributed fashion, but anonymity-preserving, peer-to-peer (P2P) address lookup is still an active research area [36], as is privacy-preserving distributed data storage in P2P systems [37]. We do not consider adversaries that operate below the application layer (e.g., by monitoring the network or even physical layer) [38], [39]. Lower-level solutions may be more appropriate against such an opponent, harnessing factors like physical proximity of users [40]. In that space, physical layer security and privacy attacks pose a very real threat, as has been documented extensively in prior work [41], [42], [43].

Organization. The remainder of this paper is organized as follows: To begin, we provide a brief summary of our results in Section II, which provides some intuition regarding the scope of the paper. We then introduce the general adaptive diffusion

TABLE I: Summary of results on the anonymity of adaptive diffusion. Here 'optimality' means adaptive diffusion has a probability of detection equal (in an order sense) to a fundamental lower bound. Shaded cells are not addressed in this work.

	Snapshot	**Spy-Based**	**Spy + Snapshot**
d-Regular Trees	Optimal (Theorem 4.1)	Asymptotically optimal in d (Theorem 5.3)	Exact probability of detection (Equation (17))
Lattice Graphs	Optimal (Proposition 4.9)		
Irregular Trees	Nearly-optimal (Theorem 4.5)	ML Estimator (Proposition 4.1 in [14])	
Social Networks	Strong anonymity (Simulations in Section IV-C2)	Strong anonymity (Simulations from Section 4 in [14])	

protocol in Section III. In Section IV, we describe how to specialize adaptive diffusion under a snapshot adversarial model. In Section V, we describe how to apply adaptive diffusion under a spy-based adversarial model. Combining the key insights of these two approaches, we introduce results from the spy+snapshot adversarial model in Section VI. For each adversarial model, we first describe the precise version of adaptive diffusion that applies to infinite d-regular trees, and show that it achieves perfect obfuscation of the source. We then provide extensions to irregular trees. We conclude by presenting simulated results over real graphs: finite, irregular, and containing cycles. In Section VII, we make a connection between adaptive diffusion on a line and Pólya's urn processes. This connection, while interesting in itself, provides a novel analysis technique for precisely capturing the price of control packets that are passed along with the messages in order to coordinate the spread of messages as per adaptive diffusion.

II. Summary

The take-home message of this paper is simple: when broadcasting a message over a fixed graph, adaptive diffusion gives strong theoretical and practical anonymity guarantees without significantly sacrificing spreading rate. We develop this thesis through a number of theorems and simulations tailored to different classes of graphs and adversarial models. The purpose of this section is to provide a high-level summary of those results, along with some intuition as to why they hold. The roadmap for this paper is subdivided by adversarial model and graph topology. Table I contains a summary of results, with links to the corresponding theorems or simulations. Note that we do not present the details of adaptive diffusion until Section III; this section makes use of only two key attributes of adaptive diffusion: (1) it breaks the symmetry of diffusion, and (2) it spreads content exponentially fast in time.

As indicated by Table I, the bulk of this paper focuses on the snapshot adversary. Mathematically, the snapshot adversary tends to be more straightforward to analyze, because it enables random spreading processes to be considered in aggregate. That is, the likelihood of a particular observed snapshot given a candidate source depends only on the spreading process' final state at the time of observation. This is in contrast with more complex adversarial models like the spy-based adversary, which provides timestamp metadata for individual nodes. Under such models, the likelihood of an observation depends on the state of the process at intermediate time steps prior to the estimation cutoff time. This causes the state space to grow combinatorially, and can complicate both estimation and analysis.

In this paper, we show that on tree-structured graphs, the likelihood of an adaptive diffusion snapshot (given a candidate source) can be computed by analyzing a random walk over the underlying graph. The length of this walk is equal to the estimation time, so its likelihood can be efficiently computed. This key observation enables a precise analysis of the ML probability of detection for adaptive diffusion on regular and irregular trees, as seen in Theorems 4.1 and 4.5, respectively. For example, on regular trees, all random walks of a fixed length are equally likely; this implies that all candidate sources are also equally likely (in reality, the proof is more nuanced, but this is a key piece of the intuition). These results can be modified to other regular graph structures, like lattices (Proposition 4.9). However, on irregular trees, different random walks have different likelihoods, which causes some nodes to be more likely sources than others. Operationally, this means that adaptive diffusion has suboptimal anonymity properties on irregular random trees; we quantify this suboptimality and propose a small modification to adaptive diffusion that empirically improves (lowers) its ML probability of detection. We validate our results on trees through simulation over a Facebook social graph dataset [44], and find empirically that our theoretical predictions extend to realistic social graphs as well (Section IV-C).

Next, we provide a brief overview of our prior results for the spy-based adversary. For general spreading models, the spy-based adversary can be difficult to analyze. However, due to the structured symmetry-breaking of adaptive diffusion, we can exactly analyze its performance against a spy-based adversary on a regular tree. This analysis is presented in [14], so we include results without proof in this paper. [14] showed that on d-regular trees, as $d \to \infty$, the ML probability of detection for adaptive diffusion approaches a fundamental lower bound. Intuitively, this occurs because adaptive diffusion is asymmetrical; by spreading messages only in some directions, the adversary loses the ability to collect independent metadata measurements from different nodes. This, in turn, lowers the adversary's ML probability of detection. The higher the degree of the underlying graph, the more pronounced this asymmetry becomes—hence, the asymptotic optimality result. Again, we confirmed our theoretical predictions for trees through simulation on a real social network; we found that the empirical performance of adaptive diffusion was very close to our fundamental lower bounds.

Finally, we consider a combination of the spy-based adversary and the snapshot adversary. Although we lack fundamental bounds on the ML probability of detection for this spy+snapshot adversary, we can precisely characterize its probability of detection, as presented in equation (17). This expression interpolates the results from the snapshot and spy-based adversaries. It shows that when the number of spies is small, a snapshot can improve the probability of detection compared to just timestamp metadata. Similarly, when the snapshot occurs at a very large estimation time, the spies' timestamp metadata can improve the probability of detection of the snapshot adversary.

In conjunction, this body of results implies that adaptive diffusion has strong theoretical and practical anonymity properties over a range of classes of graphs and adversarial models. Before exploring those properties, we begin by presenting the adaptive diffusion spreading protocol.

III. Adaptive diffusion

In this section, we describe adaptive diffusion in its most general form, and leave for later sections the specific choice of parameters involved. For the purpose of introducing adaptive diffusion, we focus specifically on an infinite d-regular tree as the underlying contact network.

We step through the intuition of the adaptive diffusion spreading model with an example, partially illustrated in Figure 2. The precise algorithm description is provided in Protocol 1. Adaptive diffusion ensures that the infected subgraph G_t at any even timestep $t \in \{2, 4, \ldots\}$ is a balanced tree of depth $t/2$, i.e. the hop distance from any leaf to the root (or the center of the graph) is $t/2$. We call the root node of G_t the "virtual source" at time t, and denote it by v_t. We use $v_0 = v^*$ to denote the true source. To keep the regular structure at even timesteps, we use the odd timesteps to transition from one regular subtree G_t to another one G_{t+2} with depth incremented by one.

More concretely, the first three steps are always the same. At time $t = 0$, the rumor source v^* selects, uniformly at random, one of its neighbors to be the virtual source v_2; at time $t = 1$, v^* passes the message to v_2. Next at $t = 2$, the new virtual source v_2 infects all its uninfected neighbors forming G_2 (see Figure 2). Then node v_2 chooses to either keep the virtual source token or to pass it along.

If v_2 chooses to remain the virtual source i.e., $v_4 = v_2$, it passes 'infection messages' to all the leaf nodes in the infected subtree, telling each leaf to infect all its uninfected neighbors. Since the virtual source is not connected to the leaf nodes in the infected subtree, these infection messages get relayed by the interior nodes of the subtree. This leads to N_t messages getting passed in total (we assume this happens instantaneously). These messages cause the rumor to spread symmetrically in all directions at $t = 3$. At $t = 4$, no spreading occurs (Figure 2, right panel).

If v_2 does *not* choose to remain the virtual source, it passes the virtual source token to a randomly chosen neighbor v_4, excluding the previous virtual source (in this example, v_0). Thus, if the virtual source moves, it moves away from the true source by one hop. Once v_4 receives the virtual source token, it sends out infection messages. However, these messages do not get passed back in the direction of the previous virtual source. This causes the infection to spread asymmetrically over only one subtree of the infected graph (G_3 in Figure 2, left panel). In the subsequent timestep ($t = 4$), the virtual source remains fixed and passes the same infection messages again. After this second round of asymmetric spreading, the infected graph is once again symmetric about the virtual source v_4 (G_4 in Figure 2, left panel).

This process continues at each timestep: the virtual source v_t chooses whether to keep or pass the virtual source token. Conditioned on this decision, the infected subgraph grows deterministically as needed to ensure symmetry about the new virtual source, v_{t+2}.

As we will see momentarily, adaptive diffusion uses varying amounts of control information to coordinate the spread of messages. In some adversarial models (snapshot), this control information does not hurt anonymity; in others (spy-based), it can be problematic. We therefore introduce different implementations of adaptive diffusion as needed, using different amounts of control information. In each implementation, the resulting distribution of the random infection process is the same (if the same parameters are used).

This random infection process can be defined as a time-inhomogeneous (time-dependent) Markov chain over the state defined by the location of the current virtual source $\{v_t\}_{t\in\{0,2,4,\ldots\}}$. By the symmetry of the underlying contact network, which we assume is an infinite d-regular tree, and the fact that the next virtual source is chosen uniformly at random among the neighbors of the current virtual source, it is sufficient to consider a Markov chain over the hop distance between the true source v^* and v_t, the virtual source at time t. Therefore, we design a Markov chain over the state

$$h_t \quad = \quad \delta_H(v^*, v_t) \ ,$$

for even t, where $\delta_H(v^*, v_t)$ denotes the hop distance between nodes v^* (the true source) and v_t (the virtual source). Figure 2 shows an example with $(h_2, h_4) = (1, 2)$ on the left and $(h_2, h_4) = (1, 1)$ on the right.

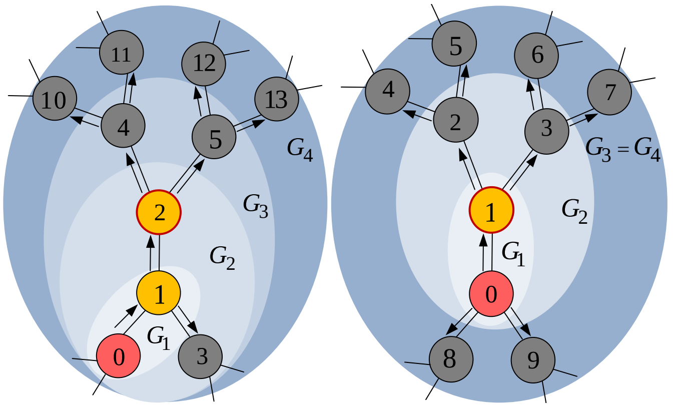


Fig. 2: Adaptive diffusion over regular trees. Yellow nodes indicate the set of virtual sources (past and present), and for $T = 4$, the virtual source node is outlined in red.

Algorithm 1 Adaptive Diffusion

```
Input: contact network $G = (V, E)$, source $v^*$, time $T$,
    degree $d$
Output: set of infected nodes $V_T$
$V_0 \leftarrow \{v^*\}$, $h \leftarrow 0$, $v_0 \leftarrow v^*$
$v^*$ selects one of its neighbors $u$ at random
$V_1 \leftarrow V_0 \cup \{u\}$, $h \leftarrow 1$, $v_1 \leftarrow u$
let $N(u)$ represent $u$'s neighbors
$V_2 \leftarrow V_1 \cup N(u) \setminus \{v^*\}$, $v_2 \leftarrow v_1$
$t \leftarrow 3$
for $t \le T$ do
    $v_{t-1}$ selects a random variable $X \sim U(0, 1)$
    if $X \le \alpha_d(t-1, h)$ then
        for all $v \in N(v_{t-1})$ do
            Infection Message($G$,$v_{t-1}$,$v$,$G_t$)
    else
        $v_{t-1}$ randomly selects $u \in N(v_{t-1}) \setminus \{v_{t-2}\}$
        $h \leftarrow h + 1$
        $v_t \leftarrow u$
        for all $v \in N(v_t) \setminus \{v_{t-1}\}$ do
            Infection Message($G$,$v_t$,$v$,$V_t$)
            if $t + 1 > T$ then
                break
            Infection Message($G$,$v_t$,$v$,$V_t$)
    $t \leftarrow t + 2$
procedure INFECTION MESSAGE($G$,$u$,$v$,$V_t$)
    if $v \in V_t$ then
        for all $w \in N(v) \setminus \{u\}$ do
            Infection Message($G$,$v$,$w$,$G_t$)
    else
        $V_t \leftarrow V_{t-2} \cup \{v\}$
```

At every even timestep, the protocol randomly determines whether to keep the virtual source token ($h_{t+2} = h_t$) or to pass it ($h_{t+2} = h_t + 1$). We specify the resulting time-inhomogeneous Markov chain over $\{h_t\}_{t \in \{2,4,6,\ldots\}}$ by choosing appropriate transition probabilities as a function of time t and current state h_t. For even t, we denote this probability by

$$\alpha_d(t, h) \quad \triangleq \quad \mathbb{P}\big(h_{t+2} = h_t | h_t = h \big) \ ,$$

where the subscript d denotes the degree of the underlying contact network. In Figure 2, at $t = 2$, the virtual source remains at the current node (right) with probability $\alpha_3(2, 1)$, or passes the virtual source to a neighbor with probability $1 - \alpha_3(2, 1)$ (left). The parameters $\alpha_d(t, h)$ fully describe the transition probability of the Markov chain defined over $h_t \in \{1, 2, \ldots, t/2\}$. For example, if we choose $\alpha_d(t, h) = 1$ for all t and h, then the virtual source never moves for $t > 1$. The message spreads almost symmetrically, so the source can be caught with high probability, much like diffusion. If we instead choose $\alpha_d(t, h) = 0$ for all t and h, the virtual source *always* moves. This ensures that the source is always at one of the leaves of the infected subgraph. We return to this special case when addressing spy-based adversaries in Section V.

The real challenge, then, is choosing the parameters $\alpha_d(t, h)$, which fully specify the virtual source transition probabilities. These parameters can significantly alter the anonymity and spreading properties of adaptive diffusion. In this work, we explain how to choose this parameter α_d to achieve desired source obfuscation.

IV. SNAPSHOT-BASED ADVERSARIAL MODEL

Under the snapshot adversarial model, an adversary observes the infected subgraph G_T at a certain time T and produces an estimate $\hat{v}$ of the source v^* of the message. Since the adversary is assumed to not have any prior information on which node is likely to be the source, we analyze the performance of the maximum likelihood estimator

$$\hat{v}_{\text{ML}} = \arg \max_{v \in G_T} \mathbb{P}(G_T | v).$$

We show that adaptive diffusion with appropriate parameters can achieve *perfect obfuscation*, i.e. the probability of detection for the ML estimator when n nodes are infected is close to $1/n$:

$$\mathbb{P}\big(\hat{v}_{\text{ML}} = v^* | N_T = n \big) \quad = \quad \frac{1}{n} + o\Big(\frac{1}{n}\Big) \ .$$

This is the best source obfuscation that can be achieved by any protocol, since there are only n candidates for the source and they are all equally likely.

A. Main Result (Snapshot Model)

In this section, we show that for appropriate choice of parameters $\alpha_d(t, h)$, we can achieve both fast spreading and perfect obfuscation over d-regular trees. We start by giving baseline spreading rates for deterministic spreading and diffusion.

Given a contact network of an infinite d-regular tree, $d > 2$, consider the following deterministic spreading protocol. At time $t = 1$, the source node infects all its neighbors. At $t \ge 2$, the nodes at the boundary of the infection spread the message to their uninfected neighbors. Thus, the message spreads one hop in every direction at each timestep. This approach is the fastest-possible spreading, infecting $N_T = 1 + d((d-1)^T - 1)/(d-2)$ nodes at time T, but the source is trivially identified as the center of the infected subtree. In this case, the infected subtree is a balanced regular tree where all leaves are at equal depth from the source.

Now consider a random diffusion model. At each timestep, each uninfected neighbor of an infected node is independently infected with probability q. In this case, $\mathbb{E}[N_T] = 1 + qd((d-1)^T - 1)/(d-2)$, and it was shown in [7] that the probability of correct detection for the maximum likelihood estimator of the rumor source is $\mathbb{P}(\hat{v}_{\text{ML}} = v^*) \ge C_d$ for some positive constant C_d that only depends on the degree d. Hence, the source is only hidden in a constant number of nodes close to the center, even when the total number of infected nodes is arbitrarily large.

Now we consider the spreading and anonymity properties of adaptive diffusion. Let $p^{(t)} = [p_h^{(t)}]_{h \in \{1,\ldots,t/2\}}$ denote the distribution of the state of the Markov chain at time t, i.e. $p_h^{(t)} = \mathbb{P}(h_t = h)$. The state transition can be represented as the following $((t/2)+1)\times(t/2)$ dimensional column stochastic matrices:

$$p^{(t+2)} = \begin{bmatrix} \alpha_d(t,1) & & & \\ 1-\alpha_d(t,1) & \alpha_d(t,2) & & \\ & 1-\alpha_d(t,2) & \ddots & \\ & & \ddots & \alpha_d(t,t/2) \\ & & & 1-\alpha_d(t,t/2) \end{bmatrix} p^{(t)}$$

We treat h_t as strictly positive, because at time $t=0$, when $h_0 = 0$, the virtual source is always passed. Thus, $h_t \geq 1$ afterwards. At all even t, we desire $p^{(t)}$ to be

$$p^{(t)} = \frac{d-2}{(d-1)^{t/2}-1} \begin{bmatrix} 1 \\ (d-1) \\ \vdots \\ (d-1)^{t/2-1} \end{bmatrix} \in \mathbb{R}^{t/2} \ , \qquad (1)$$

for $d>2$ and for $d=2$, $p^{(t)} = (2/t)\mathbf{1}_{t/2}$ where $\mathbf{1}_{t/2}$ is all ones vector in $\mathbb{R}^{t/2}$. There are $d(d-1)^{h-1}$ nodes at distance h from the virtual source, and by symmetry all of them are equally likely to have been the source:

$$\begin{aligned} \mathbb{P}(G_T|v^*, \delta_H(v^*, v_t) = h) &= \frac{1}{d(d-1)^{h-1}} p_h^{(t)} \\ &= \frac{d-2}{d((d-1)^{t/2}-1)} \ , \end{aligned}$$

for $d>2$, which is independent of h. Hence, all the infected nodes (except for the virtual source) are equally likely to have been the source of the origin. This statement is made precise in equation (4).

Together with the desired probability distribution in equation (1), this gives a recursion over t and h for computing the appropriate $\alpha_d(t,h)$'s. After some algebra and an initial state $p^{(2)} = 1$, we get that the following choice ensures the desired equation (1):

$$\alpha_d(t,h) = \begin{cases} \frac{(d-1)^{t/2-h+1}-1}{(d-1)^{t/2+1}-1} & \text{if } d>2 \\ \frac{t-2h+2}{t+2} & \text{if } d=2 \end{cases} \qquad (2)$$

With this choice of parameters, we show that adaptive diffusion spreads fast, infecting $N_t = O((d-1)^{t/2})$ nodes at time t and each of the nodes except for the virtual source is equally likely to have been the source.

Theorem 4.1: Suppose the contact network is a d-regular tree with $d \geq 2$, and one node v^* in G starts to spread a message according to Protocol 1 at time $t=0$, with $\alpha_d(t,h)$ chosen according to equation (2). At a certain time $T \geq 0$ an adversary estimates the location of the source v^* using the maximum likelihood estimator $\hat{v}_{\text{ML}}$. The following properties hold for Protocol 1:

(a) the number of infected nodes at time T is

$$N_T \geq \begin{cases} \frac{2(d-1)^{(T+1)/2}-d}{(d-2)} + 1 & \text{if } d>2 \\ T+1 & \text{if } d=2 \end{cases} \qquad (3)$$

(b) the probability of source detection for the maximum likelihood estimator at time T is

$$\mathbb{P}\left(\hat{v}_{\text{ML}} = v^*\right) \leq \begin{cases} \frac{d-2}{2(d-1)^{(T+1)/2}-d} & \text{if } d>2 \\ (1/T) & \text{if } d=2 \end{cases} \qquad (4)$$

(c) the expected hop-distance between the true source v^* and its estimate $\hat{v}_{\text{ML}}$ under maximum likelihood estimation is lower bounded by

$$\mathbb{E}[d(\hat{v}_{\text{ML}}, v^*)] \geq \frac{d-1}{d}\frac{T}{2}.$$

(Proof in Section IX-A)

Although this choice of parameters achieves perfect obfuscation, the spreading rate is slower than the deterministic spreading model, which infects $O((d-1)^T)$ nodes at time T. However, this type of constant-factor loss in the spreading rate is inevitable: the only way to deviate from the deterministic spreading model is to introduce appropriate delays.

In order to spread according to adaptive diffusion with the prescribed $\alpha_d(h,t)$, the system needs to know the degree d of the underlying contact network. However, performance is insensitive to knowledge of d for certain parameter settings, as shown in the following proposition. Specifically, one can choose $\alpha_d(h,t) = 0$ for all d, h, and t and still achieve performance comparable to the optimal choice. The main idea is that there are as many nodes in the boundary of the snapshot (leaf nodes) as there are in the interior, so it is sufficient to hide among the leaves. One caveat is that if the underlying contact network is a line (i.e. $d=2$) then this approach fails since there are only two leaf nodes at any given time, and the probability of detection is trivially $1/2$.

Proposition 4.2: Suppose that the underlying contact network G is an infinite d-regular tree with $d>2$, and one node v^* in G starts to spread a message at time $t=0$ according to Protocol 1 with $\alpha_d(h,t)=0$ for all d, h, and t. At a certain time $T \geq 1$ an adversary estimates the location of the source v^* using the maximum likelihood estimator $\hat{v}_{\text{ML}}$. Then the following properties hold:

(a) the number of infected nodes at time $T \geq 1$ is at least

$$N_T \ \geq \ \frac{(d-1)^{(T+1)/2}}{d-2} \ ; \qquad (5)$$

(b) the probability of source detection for the maximum likelihood estimator at time T is

$$\mathbb{P}(\hat{v}_{\text{ML}} = v^*) \ = \ \frac{d-1}{2+(d-2)N_T} \ ; \text{ and}$$

(c) the expected hop-distance between the true source v^* and its estimate $\hat{v}$ is lower bounded by

$$\mathbb{E}[\delta_H(v^*, \hat{v}_{\text{ML}})] \ \geq \ \frac{T}{2} \ .$$

(Proof in Section IX-B).

Multiple snapshots. The results in Theorem 4.1 and Proposition 4.2 hold for a single snapshot. However, an adversary could in principle take multiple snapshots of the same message's spread, at different points in time. We show that doing so increases the probability of detection at most by a logarithmic factor, compared to what it learns from the first snapshot (on average).

Proposition 4.3: Suppose that the underlying contact network G is an infinite d-regular tree with $d>2$, and one node v^* in G starts to spread a message at time $t=0$ according to Protocol 1, with $\alpha_d(t,h)$ chosen according to equation (2).

At a certain time $T \geq 0$ an adversary observes a snapshot G_T with N_T nodes. In timesteps $\{T_1, T_2, \ldots, T_m\}$, where $T_i > T$ for all $i \in \{1, 2, \ldots m\}$, the adversary again observes snapshots G_{T_i}. The adversary then estimates the location of the source v^* using a maximum likelihood estimator $\hat{v}_{\text{ML}}$, based on knowledge of all observed snapshots. Then the probability of source detection for the maximum likelihood estimator at time T is upper bounded as follows:

$$\mathbb{P}\big(\hat{v}_{\text{ML}} = v^*\big) \quad \leq \quad C\frac{\log_{d-1} N_T}{N_T - 1} + o\left(\frac{\log_{d-1} N_T}{N_T}\right) \quad (6)$$

where the constant C depends only only on the tree degree d. (Proof in Section IX-C).

This result suggests that an adversary cannot learn much more than the information it learns from the first snapshot; i.e., the probability of detection increases at most from $O(1/N_T)$ to $O(\log N_T / N_T)$. Moving forward, we will assume that the snapshot adversary observes only one snapshot, at time T.

B. Irregular Trees

In this section, we study adaptive diffusion on irregular trees, with potentially different degrees at the vertices. Although the degrees are irregular, we still apply adaptive diffusion with $\alpha_{d_0}(t, h)$'s chosen for a specific d_0 that might be mismatched with the graph due to degree irregularities. There are a few challenges in this degree-mismatched adaptive diffusion. First, finding the maximum likelihood estimate of the source is not immediate, due to degree irregularities. Second, it is not clear *a priori* which choice of d_0 is good. We first show an efficient message-passing algorithm for computing the maximum likelihood source estimate. Using this estimate, we illustrate through simulations how adaptive diffusion performs and show that the detection probability is not too sensitive to the choice of d_0 as long as d_0 is above a threshold that depends on the degree distribution.

Then, for the special choice of $d_0 = \infty$, we precisely characterize the maximum likelihood probability of detection and demonstrate that adaptive diffusion does not provide perfect obfuscation. Doing so requires proving a concentration result for an extreme value defined over Galton-Watson branching processes, which may be of independent interest. We use the associated analysis to propose a modification of adaptive diffusion called preferential-attachment adaptive diffusion (PAAD), which empirically improves the probability of detection over irregular trees, compared to standard adaptive diffusion.

Efficient ML estimation. To keep the discussion simple, we assume that T is even. The same approach can be naturally extended to odd T. Since the spreading pattern in adaptive diffusion is entirely deterministic given the sequence of virtual sources at each timestep, computing the likelihood $\mathbb{P}(G_T | v^* = v)$ is equivalent to computing the probability of the virtual source moving from v to v_T over T timesteps. On trees, there is only one path from v to v_T and since we do not allow the virtual source to "backtrack", we only need to compute the probability of every virtual source sequence $(v_0, v_2, \ldots, v_T)$ that meets the constraint $v_0 = v$. Due to the Markov property exhibited by adaptive diffusion, we have $\mathbb{P}(G_T | \{(v_t, h_t)\}_{t \in \{2,4,\ldots,T\}}) = \prod_{\substack{t<T-1 \\ t \text{ even}}} \mathbb{P}(v_{t+2} | v_t, h_t)$, where $h_t = \delta_H(v_0, v_t)$. For t even, $\mathbb{P}(v_{t+2} | v_t, h_t) = \alpha_d(t, h_t)$ if $v_t = v_{t+2}$ and $\frac{1-\alpha_d(t,h_t)}{d_{v_t}-1}$ otherwise. Here d_{v_t} denotes the degree of node v_t in G. Given a virtual source trajectory $\mathcal{P} = (v_0, v_2, \ldots, v_T)$, let $\mathcal{J}_{\mathcal{P}} = (j_1, \ldots, j_{\delta_H(v_0, v_T)})$ denote the timesteps at which a new virtual source is introduced, with $1 \leq j_i \leq T$. It always holds that $j_1 = 2$ because after $t = 0$, the true source chooses a new virtual source and $v_2 \neq v_0$. If the virtual source at $t = 2$ were to keep the token exactly once after receiving it (so $v_2 = v_4$), then $j_2 = 6$, and so forth. To find the likelihood of a node being the true source, we sum over *all* such trajectories

$$\mathbb{P}(G_T | v_0) = \sum_{\mathcal{J}_{\mathcal{P}} : \mathcal{P} \in \mathcal{S}(v_0, v_T, T)} \underbrace{\frac{1}{d_{v_0}} \prod_{k=1}^{\delta_H(v_0, v_T) - 1} \frac{1}{d_{v_{j_k}} - 1}}_{A_{v_0}} \times$$

$$\underbrace{\prod_{\substack{t<T \\ t \text{ even}}} \left(\mathbb{1}_{\{t+2 \notin \mathcal{J}_{\mathcal{P}}\}} \alpha_d(t, h_t) + \mathbb{1}_{\{t+2 \in \mathcal{J}_{\mathcal{P}}\}} (1 - \alpha_d(t, h_t))\right),}_{B_{v_0}} \quad (7)$$

where $\mathbb{1}$ is the indicator function and $S(v_0, v_T, T) = \{\mathcal{P} : \mathcal{P} = (v_0, v_2, \ldots, v_T)$ is a valid trajectory of the virtual source$\}$. Intuitively, part A_{v_0} of the above expression is the probability of choosing the set of virtual sources specified by $\mathcal{P}$, and part B_{v_0} is the probability of keeping or passing the virtual source token at the specified timesteps. Equation (7) holds for both regular and irregular trees. Since the path between two nodes in a tree is unique, and part A_{v_0} is (approximately) the product of node degrees in that path, A_{v_0} is identical for all trajectories $\mathcal{P}$. Pulling A_{v_0} out of the summation, we wish to compute the summation over all valid paths $\mathcal{P}$ of part B_{v_0} (for ease of exposition, we will use B_{v_0} to refer to this whole summation). Although there are combinatorially many valid paths, we can simplify the formula in equation (7) for the particular choice of $\alpha_d(t, h)$'s defined in equation (2).

Proposition 4.4: Suppose that the underlying contact network $\tilde{G}$ is an infinite tree with degree of each node larger than one. One node $\tilde{v}^*$ in $\tilde{G}$ starts to spread a message at time $t = 0$ according to Protocol 1 with the choice of $d = d_0$. At a certain even time $T \geq 0$, the maximum likelihood estimate of $\tilde{v}^*$ given a snapshot of the infected subtree $\tilde{G}_T$ is

$$\arg \max_{v \in \tilde{G}_T \setminus \tilde{v}_T} \quad \frac{d_0}{d_v} \prod_{v' \in P(\tilde{v}_T, v) \setminus \{\tilde{v}_T, v\}} \frac{d_0 - 1}{d_{v'} - 1} \quad (8)$$

where $\tilde{v}_T$ is the (Jordan) center of the infected subtree $\tilde{G}_T$, $P(\tilde{v}_T, v)$ is the unique shortest path from $\tilde{v}_T$ to v, and $d_{v'}$ is the degree of node v'.

To understand this proposition, consider Figure 3, which was spread using adaptive diffusion (Protocol 1) with a choice of $d_0 = 2$. Then equation (8) can be computed easily for each node, giving $[1/2, 1, 0, 1, 2/3, 1/2, 1/2, 1/4]$ for nodes $[1, 2, 3, 4, 5, 6, 7, 8]$, respectively. Hence, nodes 2 and 4 are most likely. Intuitively, nodes whose path to the center have small degrees are more likely. However, if we repeat this estimation assuming $d_0 = 4$, then equation (8) gives $[3, 2, 0, 2, 4/3, 3, 3, 3/2]$. In this case, nodes 1, 6, and 7 are most likely. When d_0 is large, adaptive diffusion tends to place the source closer to the leaves of the infected subtree, so leaf nodes are more likely to have been the source.

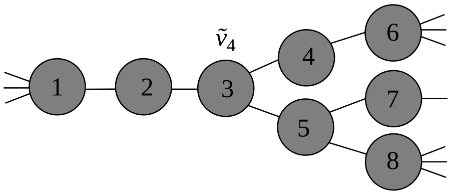


Fig. 3: Irregular tree $\tilde{G}_4$ with virtual source $\tilde{v}_4$.

Proof of Proposition 4.4: We first make two observations: (a) Over regular trees, $\mathbb{P}(G_T|u) = \mathbb{P}(G_T|w)$ for any $u \neq w \in G_T$, even if they are at different distances from the virtual souce. (b) Part B_{v_0} is identical for regular and irregular graphs, as long as the distance from the candidate source node to v_T is the same in both, and the same d_0 is used to compute $\alpha_{d_0}(t, h)$. That is, let $\tilde{G}_T$ denote an infected subtree over an *irregular* tree network, with virtual source $\tilde{v}_T$, and G_T will denote a *regular* infected subtree with virtual source v_T. For candidate sources $\tilde{v}_0 \in \tilde{G}_T$ and $v_0 \in G_T$, if $\delta_H(\tilde{v}_T, \tilde{v}_0) = \delta_H(v_T, v_0) = h$, then $B_{v_0} = B_{\tilde{v}_0}$. So to find the likelihood of $\tilde{v}_0 \in \tilde{G}_T$, we can solve for $B_{\tilde{v}_0}$ using the likelihood of $v_0 \in G_T$, and compute $A_{\tilde{v}_0}$ using the degree information of every node in the infected, irregular subgraph.

To solve for $B_{\tilde{v}_0}$, note that over regular graphs, $A_v = 1/(d_0\,(d_0-1)^{\delta_H(v,v_T)-1})$, where d_0 is the degree of the regular graph. If G is a regular tree, equation (7) still applies. Critically, for regular trees, the $\alpha_{d_0}(t,h)$'s are designed such that the likelihood of each node being the true source is equal. Hence,

$$\mathbb{P}(G_T|v_0) = \underbrace{\frac{1}{d_0(d_0-1)^{\delta_H(v_0,v_T)-1}}}_{A_{v_0}} \times B_{v_0} \ ,$$

is a constant that does not depend on v_0. This gives $B_{v_0} \propto (d_0-1)^{\delta_H(v_T,v_0)}$. From observation ($b$), we have that $B_{\tilde{v}_0} = B_{v_0}$. Thus we get that for a $\tilde{v}_0 \in \tilde{G}_T \setminus \{\tilde{v}_T\}$,

$$\begin{aligned} \mathbb{P}(\tilde{G}_T|\tilde{v}_0) &= A_{\tilde{v}_0}\, B_{\tilde{v}_0} \\ &\propto \frac{(d_0-1)^{\delta_H(\tilde{v}_T,\tilde{v}_0)}}{d_{\tilde{v}_0}\prod_{\tilde{v}' \in P(\tilde{v}_T,\tilde{v}_0)\setminus\{\tilde{v}_0,\tilde{v}_T\}}(d_{\tilde{v}'}-1)} \end{aligned}$$

After scaling appropriately and noting that $|P(\tilde{v}_T, \tilde{v}_0)| = \delta_H(\tilde{v}_T, \tilde{v}_0) + 1$, this gives the formula in equation (8). ■

We provide an efficient message passing algorithm for computing the ML estimate in equation (8), which is naturally distributed. We then use this estimator to simulate message spreading for random irregular trees and show that when d_0 exceeds a threshold (determined by the degree distribution), obfuscation is not too sensitive to the choice of d_0.

$A_{\tilde{v}_0}$ can be computed efficiently for irregular graphs with a simple message-passing algorithm. In this algorithm, each node $\tilde{v}$ multiplies its degree information by a cumulative likelihood that gets passed from the virtual source to the leaves. Thus if there are $\tilde{N}_T$ infected nodes in $\tilde{G}_T$, then $A_{\tilde{v}_0}$ for every $\tilde{v}_0 \in \tilde{G}_T$ can be computed by passing $O(\tilde{N}_T)$ messages. This message-passing is outlined in procedure 'Degree Message' of Algorithm 2. For example, consider computing A_5 for the graph in Figure 3. The virtual source $\tilde{v}_T = 3$ starts by setting $A_2 = \frac{1}{2}$, $A_4 = \frac{1}{2}$, and $A_5 = \frac{1}{3}$. This gives A_5, but to compute other values of $A_{\tilde{w}}$, the message passing continues. Each of the nodes $\tilde{v} \in N(3)$ in turn sets $A_{\tilde{w}}$ for *their* children $\tilde{w} \in N(\tilde{v})$; this is done by dividing $A_{\tilde{v}}$ by $d_{\tilde{w}}$ and replacing the factor of $\frac{1}{d_{\tilde{v}}}$ in $A_{\tilde{v}}$ with $\frac{1}{d_{\tilde{v}}-1}$. For example, node 5 would set $A_7 = \frac{A_5}{2} \cdot \frac{3}{2}$. This step is applied recursively until reaching the leaves.

Algorithm 2 Implementation of ML estimator in equation (8)

```
Input: infected network $\tilde{G}_T = (\tilde{V}_T, \tilde{E}_T)$, virtual source $\tilde{v}_T$,
    time $T$, the spreading model parameter $d_0$
Output: argmax$_{\tilde{v} \in \tilde{V}_T}$ $\mathbb{P}(\tilde{G}_T|\tilde{v}^* = \tilde{v})$
$P_{\tilde{v}} \triangleq \mathbb{P}(\tilde{G}_T|\tilde{v}^* = \tilde{v})$.
$P_{\tilde{v}_T} \leftarrow 0$
$A_{\tilde{v}} \leftarrow 1$ for $\tilde{v} \in \tilde{V}_T \setminus \{\tilde{v}_T\}$
$A_{\tilde{v}_T} \leftarrow 0$
$A \leftarrow$ Degree Message($G_T$, $\tilde{v}_T$, $\tilde{v}_T$, $A$)
$\mathbb{P}(G_T|v_{leaf}) \leftarrow \frac{1}{d_0(d_0-1)^{T/2-1}} \prod_{\substack{t<T \\ t \text{ even}}} (1-\alpha_{d_0}(t, \frac{t}{2}))\}$
for all $\tilde{v} \in \tilde{V}_T \setminus \{\tilde{v}_T\}$ do
    $h \leftarrow \delta_H(\tilde{v}, \tilde{v}_T)$
    $B_{\tilde{v}} \leftarrow \mathbb{P}(G_T|v_{leaf}) \cdot d_0 \cdot (d_0-1)^{h-1}$
    $P_{\tilde{v}} \leftarrow A_{\tilde{v}} \cdot B_{\tilde{v}}$
return argmax$_{\tilde{v} \in \tilde{V}_T} P_{\tilde{v}}$
procedure DEGREE MESSAGE($\tilde{G}_T$, $\tilde{u}$, $\tilde{v}$, $A$)
    for all $\tilde{w} \in N(\tilde{v}) \setminus \{\tilde{u}\}$ do
        if $\tilde{v} = \tilde{u}$ then
            $A_{\tilde{w}} \leftarrow A_{\tilde{v}}/d_{\tilde{w}}$
            Degree Message($\tilde{G}_T$, $\tilde{v}$, $\tilde{w}$, $A$)
        else
            if $\tilde{v}$ is not a leaf then
                $A_{\tilde{w}} \leftarrow A_{\tilde{v}} \cdot d_{\tilde{v}}/(d_{\tilde{w}} \cdot (d_{\tilde{v}}-1))$
                Degree Message($\tilde{G}_T$, $\tilde{v}$, $\tilde{w}$, $A$)
    return $A$
```

As discussed earlier, $B_{\tilde{v}_0}$ only depends on d_0 and $\delta_H(\tilde{v}_T, \tilde{v}_0)$. If $v_{\text{leaf}} \in G_T$ is a leaf node and G is a regular tree, we get

$$\mathbb{P}(G_T|v_{\text{leaf}}) = \underbrace{\frac{1}{d_0(d_0-1)^{T/2-1}}}_{A_{v_{\text{leaf}}}} \underbrace{\prod_{\substack{t<T \\ t \text{ even}}} (1-\alpha_{d_0}(t, \frac{t}{2}))}_{B_{v_{\text{leaf}}}} .$$

If $\tilde{v}_0$ is $h < T/2$ hops from $\tilde{v}_T$, then for node v_0 with $\delta_H(v_0, v_T) = h < T/2$ over a *regular* tree,

$$\mathbb{P}(G_T|v_0) = \mathbb{P}(G_T|v_{\text{leaf}}) = \frac{1}{d_0 \cdot (d_0-1)^{h-1}} B_{v_0} .$$

Finally, $B_{\tilde{v}_0} = B_{v_0}$. So to solve for B_5 in our example, we compute $\mathbb{P}(G_T|v_{leaf})$ for a 3-regular graph at time $T = 4$. This gives $\mathbb{P}(G_4|v_{leaf}) = A_{v_{leaf}} \cdot B_{v_{leaf}} = \frac{1}{6} \cdot (1-\alpha_3(2,1)) = \frac{1}{9}$. Thus $B_5 = \mathbb{P}(G_4|v_{leaf}) \cdot d_0 \cdot (d_0-1)^{h-1} = \mathbb{P}(G_4|v_{leaf}) \cdot 3 \cdot (2)^0 = \frac{1}{3}$. This gives $\mathbb{P}(\tilde{G}_4|5) = A_5 \cdot B_5 = \frac{1}{9}$. The same can be done for other nodes in the graph to find the maximum likelihood source estimate.

Simulation studies. We tested adaptive diffusion over random trees in which each node's degree was drawn i.i.d. from a fixed distribution. Figure 4 illustrates simulation results for

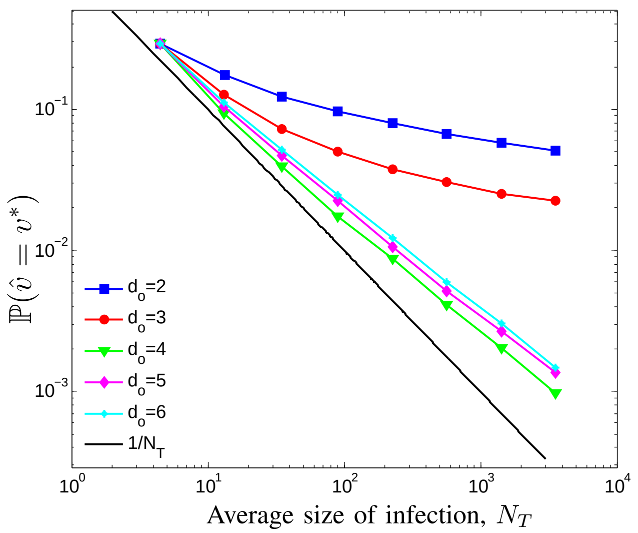


Fig. 4: The probability of detection by the maximum likelihood estimator depends on the assumed degree d_0; the source cannot hide well below a threshold value of d_0.

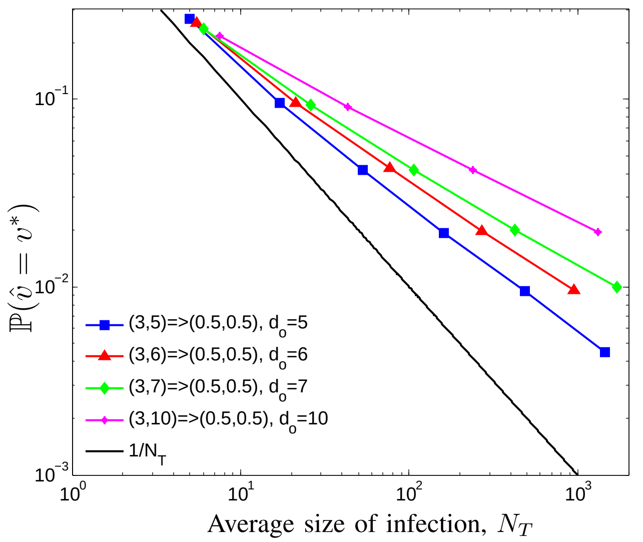


Fig. 5: Adaptive diffusion no longer provides perfect obfuscation for highly irregular graphs.

random trees in which each node has degree 3 or 4 with equal probability, averaged over 100,000 trials. By the law of large numbers, the number of nodes infected scales as $N_T \sim \mathbb{E}[D-1]^{T/2} = 2.5^{T/2}$, where D represents the degree distribution of the underlying random irregular tree. The value of d_0 corresponds to a regular tree with size scaling as $(d_0-1)^{T/2}$. Hence, one can expect that for $d_0 - 1 < 2.5$, the source is likely to be in the center of the infection, and for $d_0 > 2.45$ the source is likely to be at the boundary of the infection. Since the number of nodes in the boundary is exponentially larger than the number of nodes in the center, the detection probability is lower for $d_0 - 1 > 2.5$. This is illustrated in Figure 4, which matches our prediction. In general, choosing $d_0 = 1 + \lceil \mathbb{E}[D-1] \rceil$ provides the best obfuscation, and it is robust for values above that. In this plot, data points represent successive even timesteps; their uniform spacing on the (log-scale) horizontal axis implies the message is spreading exponentially quickly.

Figure 5 illustrates the probability of detection as a function of infection size while varying the degree distribution of the underlying tree. The notation $(3,5) => (0.5, 0.5)$ in the legend indicates that each node in the tree has degree 3 or 5, each with probability 0.5. For each distribution tested, we chose d_0 to be the maximum degree of each degree distribution. The average size of infection scales as $N_T \sim \mathbb{E}[D-1]^T$ as expected, whereas the probability of detection scales as $(d_{\min}-1)^{-T} = 2^{-T}$, which is independent of the degree distribution. This suggests that adaptive diffusion fails to provide near-perfect obfuscation when the underlying graph is irregular, and the gap increases with the irregularity of the graph. In the next section, we quantify this gap, and gain intuition about how to reduce it.

Probability of detection. In this section, we provide the probability of detection for adaptive diffusion over trees whose node degrees are drawn i.i.d. from some distribution D, for $d_0 = \infty$. However, we cannot exactly use the ML estimator from Equation 8, which assumes the infinite irregular tree G is given, and the source v^* is chosen randomly from the nodes of G. Equation 8 is the correct ML estimator in any practical scenario, but analyzing the probability of detection under this model requires a prior on the (infinitely many) nodes of G. We therefore consider a closely-related random process, in which we fix a source v^* and generate G (and consequently, G_T) on-the-fly. Specifically, at time $t = 0$, v^* draws a degree d_{v^*} from D, and generates d_{v^*} child nodes. The source picks one of these neighbors uniformly at random to be the new virtual source. Each time a node v is infected according to Protocol 1, v draws its degree d_v from D, then generates $d_v - 1$ child nodes. For example, as soon as v_2, neighbor of v^*, receives the virtual source token, it draws its degree from D and generates $d_{v_2} - 1$ children. The structure of the underlying, infinite contact network G is independent of G_T conditioned on the uninfected neighbors of the leaves of G_T, and need not be considered. The adversary observes $\mathcal{G}_T$, which is an unlabeled snapshot including G_T and its uninfected neighbors. We have that $\mathbb{P}(\hat{v}_{\text{MAP}} = v^*|T) = \sum_{\mathcal{G}_T} \mathbb{P}(\mathcal{G}_T|T)\mathbb{P}(\hat{v}_{\text{MAP}} = v^*|\mathcal{G}_T)$. We first consider $\mathbb{P}(\hat{v}_{\text{ML}} = v^*|\mathcal{G}_T)$.

1) Probability of Detection Given a Snapshot: The adversary observes this random process at time T (i.e., it observes $\mathcal{G}_T$, knowing that the interior G_T are the infected nodes), and estimates one of the leaf node as an estimate of the true source which started the random process. The following theorem analyzes the probability of detecting the true source for any estimate $\hat{v}$, given a snapshot $\mathcal{G}_T$.

Theorem 4.5: Under the above described random process of adaptive diffusion, an adversary observes the snapshot $\mathcal{G}_T$ at an even time $T > 0$ and estimates $\hat{v} \in \partial G_T$. For any estimator $\hat{v}$, the conditional probability of detection is

$$\mathbb{P}(\hat{v} = v^*|\mathcal{G}_T) \;=\; \frac{1}{d_{v_T}} \prod_{\substack{w \in \phi(\hat{v}, v_T) \\ \setminus \{v_T, \hat{v}\}}} \frac{1}{(d_w - 1)} \,, \qquad (9)$$

where v_T is the center of $\mathcal{G}_T$, $\phi(\hat{v}, v_T)$ is the (unique) path from $\hat{v}$ to v_T, G_T is the interior of $\mathcal{G}_T$ which is the infected sub-tree, and ∂G_T is the set of leaves of G_T.

A proof is provided in Section IX-D. Intuitively, equation (9) is the probability that the virtual source starting from $\hat{v}$ ends up at v_T (up to some constant factor for normalization). This gives a simple rule for the adversary to achieve the best detection probability by computing the MAP estimate:

$$\hat{v}^{(T)}_{\text{MAP}} \;\in\; \arg\max_{\hat{v}} \; \mathbb{P}(\hat{v}^{(T)} = v^* | \mathcal{G}_T) \; . \qquad (10)$$

Corollary 4.6: Under the hypotheses of Theorem 4.5, the MAP estimator in equation (10) can be computed as

$$\hat{v}^{(T)}_{\text{MAP}} \;=\; \arg \min_{v \in \partial G_T} \prod_{\substack{w \in \phi(v, v_T) \\ \setminus \{v_T, v\}}} (d_w - 1) \; , \qquad (11)$$

achieving a conditional probability of detection

$$\mathbb{P}(\hat{v}^{(T)}_{\text{MAP}} = v^* | \mathcal{G}_T) \;=\; \max_{v \in \partial G_T} \frac{1}{d_{v_T} \prod_{\substack{w \in \phi(v, v_T) \\ \setminus \{v_T, v\}}} (d_w - 1)} . \qquad (12)$$

When applied to regular trees, this recovers known results of [45], which confirms that adaptive diffusion provides strong anonymity guarantees under d-regular trees. But more importantly, Corollary 4.6 characterizes how the anonymity guarantee depends on the general topology of the snapshot. We illustrate this in two extreme examples: a regular tree and an extreme example in Figure 6.

For a d-regular tree, where all nodes have the same degree, the size of infection at even time T is the number of nodes in a d-regular tree of depth $T/2$:

$$N_T \;=\; \frac{d(d-1)^{T/2}}{d-2} + \frac{2}{d-2} \; .$$

To achieve a perfect obfuscation, we want the probability of detection to decay as $1/N_T$. We can apply Corollary 4.6 to this d-regular tree and show the probability of detection is $((d-1)/d)(d-1)^{-T/2})$, which recovers one of the known results in [45, Proposition 2.2]. This confirms that adaptive diffusion achieves near-perfect obfuscation, up to a small factor of $(d-1)/(d-2)$.

On the other hand, when there exists a path to a leaf node consisting of low-degree nodes, adaptive diffusion can be sub-optimal, and the gap to optimality can be made arbitrarily large. Figure 6 illustrates such an example. This is a tree where all nodes have the same degree $d = 5$, except for those nodes along the path from the center v_T to a leaf node v, including v_T and excluding v. The center v_T has degree two and the nodes in the path have degree three. Hence, the shaded triangles indicate d-regular subtrees of appropriate heights. The size of this infection is $N_T = ((d-1)^{T/2+1}/(d-2)^2)(1 + o(1))$. Ideally, one might hope to achieve a probability of detection that scales as $1/(d-1)^{T/2}$. However, Corollary 4.6 shows that the adaptive diffusion achieves probability of detection $1/2^{T/2}$, with the leaf node v achieving this maximum in equation (12). Hence, there is a multiplicative gap of $((d-1)/2)^{T/2}$. By increasing d, the gap can be made arbitrarily large. On the other hand, such an extreme topology is rare under the i.i.d tree model.

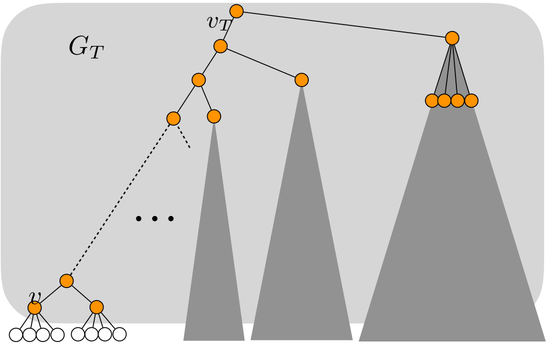


Fig. 6: An example of a snapshot emphasizing the sub-optimality of adaptive diffusion.

2) Concentration of Probability of Detection: Depending on the topology, adaptive diffusion can be significantly sub-optimal. A natural question is "what is the typical topology of a graph resulting from the random tree model?" Under the model introduced previously, we give a concrete answer. Perhaps surprisingly, this typical topology can be characterized by solving a simple convex optimization.

We are interested in the following extremal value

$$\Lambda_{G_T} \;\equiv\; d_{v_T} \min_{v \in \partial G_T} \prod_{\substack{w \in \phi(v, v_T) \\ \setminus \{v_T, v\}}} (d_w - 1) \; ,$$

which captures the topology of the snapshot. We want to characterize the typical value of this function over random tree G_T resulting from the adaptive diffusion process.

Observe that the distribution of the balanced tree G_T follows a simple branching process known as Galton-Watson process. This is because G_T resulting from adaptive diffusion has the same distribution, independent of the location of the source v^*. We consider a given degree distribution D. We use D to denote both a random variable and its distribution—the distinction should be clear from context. The random variable D has support $\boldsymbol{f} = (f_1, \dots, f_\eta)$ associated with probability $\boldsymbol{p} = (p_1, \dots, p_\eta)$ such that the degree of node v is i.i.d. with

$$d_v \;=\; \begin{cases} f_1 & \text{with probability } p_1 \; , \\ \;\vdots & \;\vdots \\ f_\eta & \text{with probability } p_\eta \; , \end{cases} \qquad (13)$$

where $2 < f_1 < f_2 < \cdots < f_\eta$ are integers and the positive p_i's sum to one. We also assume D's support set has at least two elements, i.e., $\eta \geq 2$.

Note that the adaptive diffusion always passes the virtual source token to a uniformly-chosen neighbor. It is straightforward to show that adaptive diffusion starting from a leaf node v^* has the same distribution over graphs as the following branching process, denoted $\overline{G}_T$: at time $T = 0$ a root node, which we denote as the virtual source v_T, creates D offspring. At each subsequent even time step, each leaf node in G_T creates new offspring independently according to $D-1$ (where we subtract one because each leaf is already connected to its parent). This process is repeated until time step T, which generates a random tree G_T. More precisely, the two branching processes are equal in distribution: $\overline{G}_T \overset{D}{=} G_T$. This can be seen by observing that conditioned on the path of nodes $\phi(v^*, v_T)$, the branching processes are identical. Since the node degrees in this path are drawn independently, the path is equally distributed whether it starts from the virtual source v_T or the leaf node v^*.

The following theorem provides a concentration inequality on the extremal quantity Λ_{G_T}, which in turn determines the probability of detection as provided by Corollary 4.6:

$$\mathbb{P}(\hat{v}_{\text{MAP}}^{(T)} = v^* | G_T) \quad = \quad \frac{1}{\Lambda_{G_T}} \,. \tag{14}$$

Theorem 4.7: For an even $T > 0$, suppose a random tree G_T is generated from the root v_T according to the Galton-Watson process with i.i.d. degree distribution D, where $\boldsymbol{f}$ and $\boldsymbol{p}$ are defined as in equation (13), then the following results hold:

(a) If $p_1(f_1 - 1) > 1$, for any positive $\delta > 0$, there exists positive constants $C_{D,\delta}$ and $C'_{D,\delta}$ that depend only on the degree distribution and the choice of δ such that

$$\mathbb{P}\left(\left| \frac{\log(\Lambda_{G_T})}{T/2} - \log(f_1 - 1) \right| > \delta \right) \leq e^{-C_{D,\delta} T} \,,$$

for an even time $T \geq C'_{D,\delta}$.

(b) If $p_1(f_1 - 1) < 1$, define the mean number of children:

$$\mu_D \;\equiv\; \sum_{i=1}^{\eta} p_i(f_i - 1) \,,$$

and the set

$$\mathcal{R}_D \;=\; \left\{ \boldsymbol{r} \in S_\eta \;|\; \log(\mu_D) \geq D_{\text{KL}}(\boldsymbol{r} \| \boldsymbol{\beta}) \right\} ,$$

where S_η denotes the η-dimensional probability simplex, $D_{\text{KL}}(\cdot\|\cdot)$ denotes Kullback-Leibler divergence, and $\boldsymbol{\beta}$ is a length-η probability vector in which $\beta_i = p_i(f_i - 1)/\mu_D$. Further, define $\boldsymbol{r}^*$ as follows:

$$\boldsymbol{r}^* = \quad \underset{\boldsymbol{r} \in \mathcal{R}_D}{\arg\min} \quad \langle\, \boldsymbol{r} \,,\, \log(\boldsymbol{f} - 1) \,\rangle \,, \tag{15}$$

where $\langle \boldsymbol{r}, \log(\boldsymbol{f} - 1) \rangle = \sum_{i=1}^{\eta} r_i \log{(f_i - 1)}$. Then for any $\delta > 0$, there exists positive constants $C_{D,\delta}$ and $C'_{D,\delta}$ that only depend on the degree distribution D and the choice of $\delta > 0$ such that

$$\mathbb{P}\left(\left| \frac{\log(\Lambda_{G_T})}{T/2} - \langle \boldsymbol{r}^*, \log(\boldsymbol{f} - 1) \rangle \right| > \delta \right) \;\leq\; e^{-C_{D',\delta} T}$$

for an even time $T \geq C'_{D,\delta}$.

The results in parts (a) and (b) can be merged, in the sense that the solution of equation (15) is $\boldsymbol{r}^* = [1, 0, \ldots, 0]$ when $p_1(f_1 - 1) > 1$. A proof of this theorem is provided in Section IX-E. Putting it together with equation (14), it follows that the probability of detection concentrates around

$$-\frac{2}{T} \log\left(\mathbb{P}(\hat{v}_{\text{MAP}}^{(T)} = v^*) \right) \quad \simeq \quad \langle \boldsymbol{r}^* \,,\, \log(\boldsymbol{f} - 1) \rangle \,,$$

in case (b) and around $\log(f_1 - 1)$ in case (a). Here $\simeq$ indicates concentration for large enough T. We want to emphasize that $\boldsymbol{r}^*$ can be computed using off-the-shelf optimization tools, since the program in equation (15) is a convex program of dimension η. This follows from the fact that the objective is linear in $\boldsymbol{r}$ and the feasible region is convex since KL divergence is convex in $\boldsymbol{r}$.

For example, if D is 3 w.p. 0.7 or 4 w.p. 0.3, then this falls under case (a). The theorem predicts the probability of detection to decay as $(3-1)^{-T/2}$. On the other hand, if

$$D = \begin{cases} 2 & \text{with probability } 0.3 \\ 3 & \text{with probability } 0.7 \end{cases} ,$$

then this falls under case (b) with $\mu_D = 1.7$, $\beta_1 = 0.3/1.7$, and $\beta_2 = 1.4/1.7$. In this case, the exponent is a solution of the following optimization for $\boldsymbol{r} = [r, 1 - r]$:

$$\begin{array}{ll} \underset{r \in \mathbb{R}}{\text{minimize}} & r \log 1 + (1 - r) \log 2 \\ \text{subject to} & r \log \dfrac{1.7r}{0.3} + (1 - r) \log \dfrac{1.7(1 - r)}{1.4} \leq \log(1.7) \\ & r \in [0, 1] \end{array}$$

It follows that the optimal solution is $\boldsymbol{r}^* \simeq [0.64,\ 0.36]$ and the probability of detection decays as $2^{-0.36(T/2)}$. Figure 7 confirms this prediction with simulations for these examples.

Theorem 4.7 provides a simple convex program that computes the probability of detection for any degree distribution. For random trees, this quantifies the gap between what adaptive diffusion can guarantee and the perfect obfuscation one desires. We define the rescaled log-multiplicative gap as

$$\Delta_D \quad \equiv \quad \frac{2}{T} \log \frac{\mathbb{P}(v_{\text{MAP}}^{(T)} = v^*)}{1/\mathbb{E}[|\partial G_T|]} \,,$$

where $|\partial G_T|$ is the total number of candidates in a snapshot. It is not difficult to show that $\mathbb{E}[|\partial G_T|] = \mu_D^{T/2}$, and it follows that $\Delta_D \simeq \log \mu_D - \langle \boldsymbol{r}^*, \log(\boldsymbol{f} - 1) \rangle$. For example, $\Delta_D = 0$ for regular trees, and $\Delta_D = \log_2 2.3 - \log_2 2 = 0.20$ for the first example under case (a) and $\Delta_D = \log_2 1.7 - 0.36 = 0.41$ for the second example under case (b).

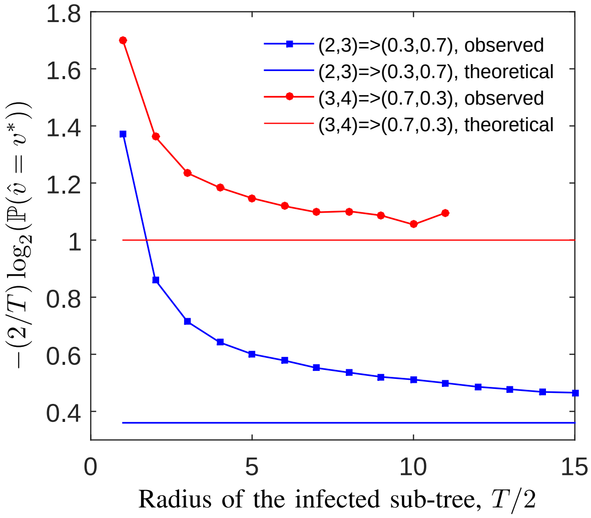


Fig. 7: Empirical verification of Theorems 4.5 and 4.7. We observe that the probability of detection converges in time to the predicted values, which depend only on the underlying degree distribution.

Simulation studies. Figure 7 empirically checks the predictions in Theorems 4.5 and 4.7. The distribution with support

$\boldsymbol{f} = (3,4)$ with probabilities $\boldsymbol{p} = (0.5, 0.5)$ addresses case 1 from the theorem, where $p_1(f_1 - 1) > 1$. The distribution with support $\boldsymbol{f} = (2,3)$ with probabilities $\boldsymbol{p} = (0.3, 0.7)$ addresses case 2, where $p_1(f_1 - 1) < 1$. In both examples, we observe that the empirical $\log(\mathbb{P}(\hat{v} = v^*))/(T/2)$ converges to the theoretical value predicted in Figure 7. However, this convergence may be slow, and the timestep duration of these experiments was limited by computational considerations since the graph size grows exponentially in time.

3) Preferential Attachment : Our analysis reveals that adaptive diffusion can be significantly sub-optimal, when the underlying graph degrees are highly irregular. To bridge this gap, we introduce a family of protocols we call *Preferential Attachment Adaptive Diffusion (PAAD)*. We analyze the performance of PAAD and provide numerical simulations showing that PAAD improves over adaptive diffusion when degrees are irregular.

The reason for this gap is that in typical random trees, there are nodes that are significantly more likely to be the source, compared to other typical candidate nodes. To achieve near-perfect obfuscation, we want all candidate nodes to have similar posterior probabilities of being the source. To balance the posterior probabilities of leaf nodes, we suggest passing the virtual source with higher probability to high-degree nodes. We propose a family of protocols based on this idea, and make this intuition precise in Theorem 4.8.

PAAD is based on adaptive diffusion, but we modify how virtual sources are chosen. We parametrize this family of protocols by a non-negative integer g. When a new virtual source is to be chosen, instead of choosing uniformly among its neighbors (except for the previous virtual source), the new virtual source is selected with probability weighted by the size of its g-hop neighborhood. Let $\mathcal{N}_g(v)$ denote the set of g-hop neighbors of node v, and let $\mathcal{N}_g(v, w)$ denote the same set, removing any nodes z for which $w \in \phi(z, v)$, where $\phi(z, v)$ denotes the path between z and v. Then for instance, if $g = 1$, then each time the virtual source is passed from v_T to v_{T+2}, it is passed to a neighbor $w \in \mathcal{N}_1(v_T, v_{T-2})$ with probability proportional to $d_w - 1$:

$$\mathbb{P}(v_{T+2} = w) = \frac{d_w - 1}{\sum_{w' \in \mathcal{N}_1(v_T, v_{T-2})}(d_{w'} - 1)} \ .$$

For general g, the probability is proportional to the size of the candidate w's g-hop local neighborhood, excluding those in the direction of the current virtual source v_T. Each virtual source v_T chooses the next virtual source as follows: for any node $w \in \mathcal{N}_1(v_T, v_{T-2})$,

$$P(v_{T+2} = w) = \frac{|\mathcal{N}_g(w, v_T)|}{\sum_{w' \in \mathcal{N}_1(v_T, v_{T-2})} |\mathcal{N}_g(w', v_T)|} \ .$$

PAAD encourages the virtual source to traverse high-degree nodes. This balances the posterior probabilities, by strengthening the probability of leaf nodes whose path contain high-degree nodes, while weakening those with low-degree nodes.

This intuition is made precise in the following theorem, which analyzes the probability of detection for a given snapshot. Define the probability that the sequence of decisions on choosing the virtual sources results in the path from a source v to the current virtual source v_T as $Q(\mathcal{G}_T, v) \equiv \prod_{t=1}^{T/2} \mathbb{P}(v_{2t} = w_t)$, where $\phi(v, v_T) = (w_0 = v, w_1, w_2, \ldots, w_{T/2-1}, w_{T/2} = v_T)$. The specific probability depends on the choice of g and the topology of the underlying tree. Note that the progression of the virtual source now depends on g-hop neighborhood, and we therefore define $\mathcal{G}_T$ to include the current infected subgraph G_T and its $(g+1)$-hop neighborhood.

Theorem 4.8: Suppose a node v^* starts to spread a message at time $t = 0$ according to PAAD, where the underlying irregular tree is generated according to the random branching process described in Section IV-B. At a certain even time $T \geq 0$, an adversary observes the snapshot of the infected subtree $\mathcal{G}_T$ and computes a MAP estimate of the source v^*. Then, the following results hold:

(a) The MAP estimator is

$$\hat{v}_{\text{MAP}} \ = \ \arg \max_{v \in \partial G_T} \ d_v \, Q(\mathcal{G}_T, v)$$

where ∂G_T denotes the leaves of G_T.

(b) The conditional probability of detection achieved by the MAP estimator is

$$\mathbb{P}(\hat{v}_{\text{MAP}} = v^* | \mathcal{G}_T) \ = \ \frac{\max_{v \in \partial G_T} d_v \, Q(\mathcal{G}_T, v)}{\sum_{w \in \partial G_T} d_w \, Q(\mathcal{G}_T, w)}$$

The proof relies on the techniques developed for Theorem 4.5, and is omitted due to space limitation. The example from Figure 6 illustrates the power of PAAD. For this class of snapshots, it is straightforward to show that under adaptive diffusion, $P_D^{AD} = 2^{-T/2}$, whereas under 1-hop PAAD,

$$P_D^{PAAD} \leq \frac{2}{(d-1)^{T/2-1} - 1} .$$

Notice from these expressions that P_D^{PAAD} scales as $(d-1)^{-T/2}$, which achieves perfect obfuscation, whereas regular adaptive diffusion scales as $2^{-T/2}$.

This shows that there exist snapshots where PAAD significantly improves over adaptive diffusion. However, such examples are rare under the random tree model, and there are also examples of snapshots where adaptive diffusion can achieve a better obfuscation than PAAD. To complete the analysis, we would like to show the analog of Theorem 4.7 for PAAD. However, the observed snapshot is no longer generated by a standard Galton-Watson branching process, due to the preferential attachment. The analysis techniques developed for Theorem 4.7 do not generalize, and new techniques seem to be needed for a technical analysis. This is outside the scope of this manuscript, but we show simulations suggesting that PAAD improves over adaptive diffusion.

Simulation studies. PAAD requires each virtual source to know some information about its local neighborhood on the contact network; in exchange, we observe empirically that it hides the source better than traditional adaptive diffusion. Figure 8 shows the probability of detection over graphs with a degree distribution of support $\boldsymbol{f} = (2,5)$ with probability $\boldsymbol{p} = (0.5, 0.5)$. The results are averaged over 10,000 realizations of the random graph and the spreading sequence. This plot shows empirically that preferential attachment adaptive

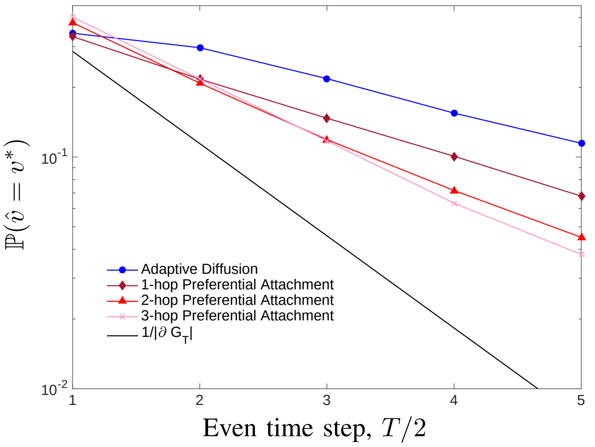


Fig. 8: Probability of detection of regular adaptive diffusion compared to 1-, 2-, and 3-hop preferential attachment adaptive diffusion (PAAD).

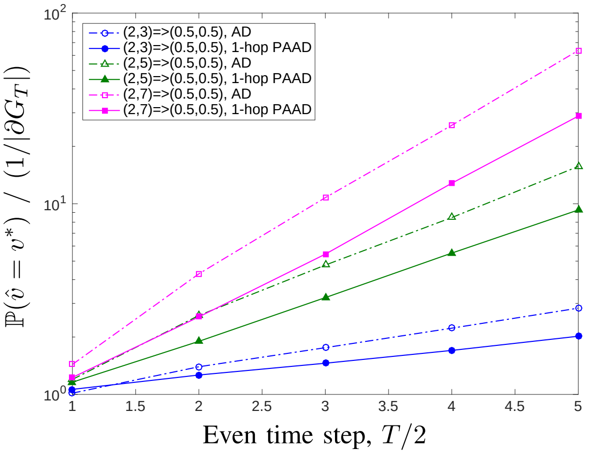


Fig. 9: Ratio of observed probability of detection to lower-bound probability of detection, for a range of degree distributions. PAAD has better anonymity properties than regular adaptive diffusion over random, irregular trees.

diffusion exhibits better hiding properties than regular adaptive diffusion, and that the benefit of preferential attachment increases with the size of the neighborhood considered for preferential attachment (e.g., one-hop vs. two-hop). Notice that our lower bound on probability of detection is $1/|\partial G_T|$ rather than $1/N_T$, as in [45]; this is because we constrain the source to always be at one of the leaves of the graph, so $1/|\partial G_T|$ lower bounds the probability of detection.

Figure 9 computes the ratio of the observed probability of detection to a lower bound on the probability of detection (i.e., $1/|\partial G_T|$), for both adaptive diffusion (AD) and one-hop PAAD. Empirically, we observe that the advantage of PAAD is greater when the degree distribution is more imbalanced (i.e., when $f_{\max} - f_{\min}$ is large).

C. General Graphs

In this section, we demonstrate how adaptive diffusion fares over graphs that involve cycles, irregular degrees, and finite graph size. We provide theoretical guarantees for the special case of two-dimensional grid graphs, and we show simulated results over a social graph dataset.

1) Grid graphs: Here, we derive the optimal parameters $\alpha(t, h)$ for spreading with adaptive diffusion over an infinite *grid graph*, defined as the graph Cartesian product of two infinite line graphs. This example highlights challenges associated with spreading over cyclic graphs, while still providing a regular, symmetric structure. To spread over grids, we make some changes to the adaptive diffusion protocol, outlined in Protocol 5 (grid adaptive diffusion).

First, standard adaptive diffusion requires the virtual source to know its distance from the true source. Over trees, this information was transmitted by passing a distance counter, h_t, that was incremented each time the virtual source changed; since the network was a tree, this distance from the source was non-decreasing as long as the virtual source was non-backtracking. However, on a cyclic graph (e.g., a grid), the virtual source's non-backtracking random walk could actually cause its distance from the true source to *decrease* with time. We wish to avoid this to preserve adaptive diffusion's anonymity guarantees.

Therefore, instead of passing the raw hop distance h_t to each new virtual source, grid adaptive diffusion passes *directional* coordinates (h_t^H, h_t^V) detailing the virtual source's horizontal and vertical displacement from the source, respectively. For example, in Figure 10, the virtual source v_4 would receive parameters $(h_t^H, h_t^V) = (-1, 1)$ because it is one hop west and one hop north of the true source. This indexing assumes some notion of directionality over the underlying contact network; nodes should know whether they received a message from the north, south, east, or west. If a virtual source chooses to move, it always passes the token to a node that is further away from the true source, i.e. $|h_{t+1}^H| + |h_{t+1}^V| \geq |h_t^H| + |h_t^V|$.

To maintain symmetry about the virtual source, we also modify the message-passing algorithm. Just as in adaptive diffusion over trees, when a *new* virtual source sends out branching messages, it sends them in every direction except that of the old virtual source. However, unlike adaptive diffusion over trees, each branch message has up to two "forbidden" directions: the direction of the previous virtual source, and the direction of the node that originated the branching message (these might be the same). Thus, if a branch message is sent west, and the previous virtual source was south of the current virtual source, each node would *only* propagate the message west and/or north. Whenever a node receives a branch message and its neighbors are not all infected, it infects all uninfected neighbors. As in adaptive diffusion over trees, two waves of directional branching messages are sent each time the virtual source moves, in every direction except that of the old virtual source. If the virtual source instead chooses to stay fixed, then the same rules hold, except the new virtual source only sends one wave of branch messages, symmetrically in every direction.

Given the spreading protocol, we can choose $\alpha(t, h)$ to give optimal hiding:

$$\alpha(t, h) = \frac{t - 2(h - 1)}{t + 4}. \tag{16}$$

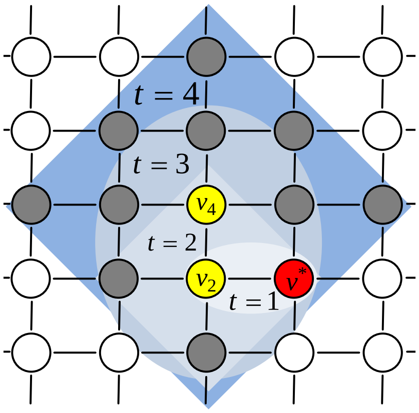


Fig. 10: Grid adaptive diffusion spreading pattern.

Under these conditions, the following result shows that we achieve perfect obfuscation, i.e. $\mathbb{P}(\hat{v}_{\text{ML}} = v^*) = 1/N_T + o(1/N_T)$.

Proposition 4.9: Suppose the contact network is an infinite grid, and one node v^* in G starts to spread a message according to Protocol 5 (grid adaptive diffusion) at time $t = 0$, with $\alpha(t, h)$ chosen according to equation (16). At a certain time $T \geq 0$ an adversary estimates the location of the source v^* using the maximum likelihood estimator $\hat{v}_{\text{ML}}$. The following properties hold for Protocol 5:

(a) the number of infected nodes at time T is

$$N_T \geq \frac{(T+1)^2}{2}$$

(b) the probability of source detection for the maximum likelihood estimator at time T is

$$\mathbb{P}\left(\hat{v}_{\text{ML}} = v^*\right) \leq \frac{2}{(T+3)(T-1)}.$$

(Proof in Section IX-F)

The baseline infection rate for deterministic, symmetric spreading is $N_T = T^2 + (T+1)^2$. Grid adaptive diffusion infection rate is within a constant factor of this maximum possible rate, and it achieves perfect obfuscation over grid graphs. The price to pay for this non-tree graph is that (a) a significant amount of metadata needs to be transmitted to coordinate the spread—particularly with respect to the directionality of messages; and (b) the position of the nodes w.r.t. a global reference needs to be known. Hence, the current implementation of the grid adaptive diffusion has a limited scope, and it remains an open question how to avoid such requirements for grids and still achieve a perfect obfuscation.

2) Real-world social graphs: In this section, we provide simulation results from running adaptive diffusion over an underlying connectivity network of 10,000 Facebook users, as described by the Facebook WOSN dataset [44]. We eliminated all nodes with fewer than three friends (this approach is taken by several existing anonymous applications so users cannot guess which of their friends originated the message), which left us with a network of 9,502 users.

Over this underlying network, we selected a node uniformly at random as the rumor source, and spread the message using adaptive diffusion for trees. We did not use grid adaptive diffusion because Protocol 5 assumes the underlying graph has a symmetric structure with a global notion of directionality, whereas the tree-based adaptive diffusion makes no such assumptions. We set $d_0 = \infty$, which means that the virtual source is always passed to a new node (i.e., $\alpha_d(t, h) = 0$). This choice is to make the ML source estimation faster; other choices of d_0 may outperform this naive choice. To preserve the symmetry of our constructed trees as much as possible, we constrained each infected node to infect a maximum of three other nodes in each timestep. We also give the adversary access to the undirected infection *subtree* that explicitly identifies all pairs of nodes for which one node spread the infection to the other. This subtree is overlaid on the underlying contact network, which is not necessarily tree-structured. We demonstrate in simulation (Figure 11) that even with this strong side information, the adversary can only identify the true message source with low probability.

Using the naive method of enumerating every possible message trajectory, it is computationally expensive to find the exact ML source estimate since there are 2^T possible trajectories, depending on whether the virtual source stayed or moved at each timestep. If the true source is one of the leaves, we can closely approximate the ML estimate *among all leaf nodes*, using the same procedure as described in IV-B, with one small modification: in graphs with cycles, the term $(d_{v_{j_k}} - 1)$ from equation (7) should be substituted with $(d^u_{v_{j_k}} - 1)$, where $d^u_{v_{j_k}}$ denotes the number of uninfected neighbors of v_{j_k} at time j_k. Loops in the graph cause this value to be time-varying, and also dependent on the location of v_0, the candidate source. This approach is only an approximation of the ML estimate because the virtual source could move in a loop over the social graph (i.e., the same node could be the virtual source more than once, in nonadjacent timesteps). We did not approximate the ML estimate for non-leaves because the simplifications used in Section IV-B to compute the likelihood no longer hold, leading to an exponential increase in the problem dimension.

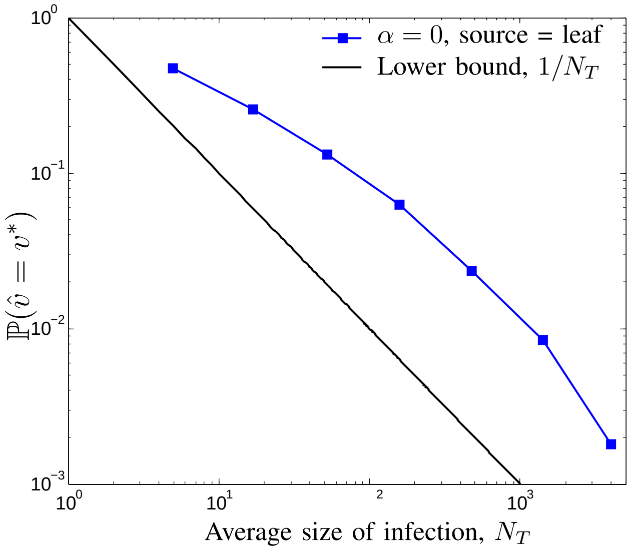


Fig. 11: Near-ML probability of detection for the Facebook graph with adaptive diffusion.

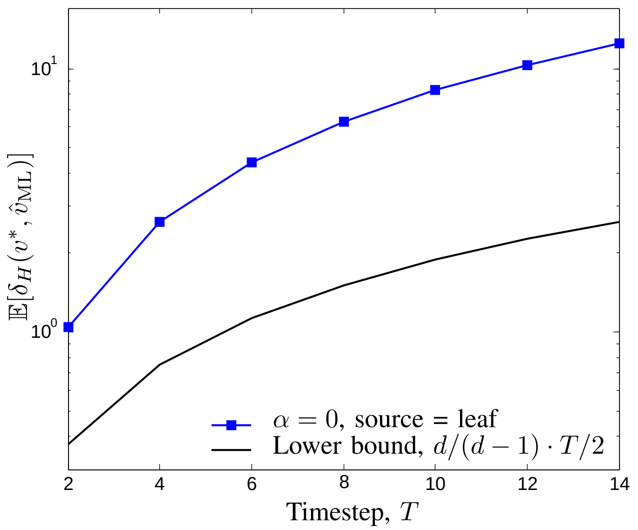


Fig. 12: Hop distance between true source and estimated source over infection subtree for adaptive diffusion over the Facebook graph.

On average, adaptive diffusion reached 96 percent of the network within 10 timesteps using $d_0 = 4$. We also computed the average distance of the true source from the estimated source *over the infected subtree* (Figure 12). We see that as time progresses, so does the hop distance of the estimated source from the true source. In social networks, nearly everyone is within a small number of hops (say, 6 hops [46]) from everyone else, so this computation is not as informative in this setting. However, it is relevant in location-based connectivity graphs, which can induce large hop distances between nodes.

V. Spy-based adversarial model

The spy-based adversary collects more detailed information than the snapshot adversary, but only for a subset of network nodes. In this section, we provide some results stating that over d-regular trees, choosing $\alpha_d(t,h) = 0$ gives asymptotically optimal hiding in d. While the proofs for these results are not included in this paper (all proofs can be found in [14]), the results are included for completeness.

For the spy-based adversary, we model each node other than the source as a spy with probability p. At some point in time, the source node v^* starts propagating its message over the graph according to some spreading protocol (e.g., diffusion or adaptive diffusion). Each spy node $s_i \in V$ observes: (1) the time T_{s_i} (relative to an absolute reference) at which it receives the message, (2) the parent node p_{s_i} that relayed the message, and (3) any other metadata used by the spreading mechanism (such as control signaling in the message header). At some time, spies *aggregate* their observations; using the collected metadata and the structure of the underlying graph, the adversary estimates the author of the message, $\hat{v}$.

To define perfect obfuscation for this adversarial model, we first observe the following:

Proposition 5.1 ([14]): Under a spy-based adversary, no spreading protocol can have a probability of detection less than p.

This results from considering the *first-spy estimator*, which returns the parent of the first spy to observe the message. Regardless of spreading, this estimator returns the true source with probability at least p; with probability p, the first node (other than the true source) to see the message is a spy.

We therefore say a protocol achieves *perfect obfuscation* against a spy-based adversary if the ML probability of detection conditioned on the spy probability p is bounded by

$$\mathbb{P}\big(\,\hat{v}_{\text{ML}} = v^* | p\big) \quad = \quad p + o\Big(p\Big)\,.$$

However, when the underlying graph is a d-regular tree, the probability of detection increases over time for standard diffusion spreading, since the estimator receives more information. Moreover, it is straightforward to show that the probability of detection tends to 1 as degree of the underlying graph $d \to \infty$:

Proposition 5.2 ([14]): Suppose the contact network is a regular tree with degree d. There is a source node v^*, and each node other than the source is chosen to be a spy node i.i.d. with probability p as described in the spy model. In each timestep, each infected node infects each uninfected neighbor independently with probability q. Then the probability of detection $\mathbb{P}(\hat{v}_{\text{ML}} = v^*) \geq 1 - (1-qp)^d$.

This bound implies that as degree increases, the probability of detecting the true source of diffusion approaches 1. The proposition also results from analyzing the first-spy estimator. These observations suggest that diffusion provides poor anonymity guarantees in real networks; contact networks may be high degree, and the adversary is not time-constrained.

A. Main result (Spy-based adversary)

In this section, we give results stating that over d-regular trees, adaptive diffusion with $\alpha_d(t,h) = 0$ achieves asymptotically perfect obfuscation in d. We also show that adaptive diffusion hides the source better than diffusion over d-regular trees, $d > 2$. However, these results depend on a slightly modified implementation of adaptive diffusion, in which some additional metadata is passed around. This implementation, which we call the Tree Protocol, facilitates analysis and is also fully distributed, avoiding the explicit notion of a virtual source.

Tree Protocol. The spreading protocol follows Algorithm 1 (Spreading on a tree) from [14]; the goal is to build an infected subtree with the true source at one of the leaves. Whenever a node v passes a message to node w, it includes three pieces of metadata: (1) the *parent node* $p_w = v$, (2) a binary *direction* indicator $u_w \in \{\uparrow, \ \downarrow\}$, and (3) the node's *level* in the infected subtree $m_w \in \mathbb{N}$. The parent p_w is the node that relayed the message to w. The direction bit u_w flags whether node w is a *spine* node, responsible for increasing the depth of the infected subtree. The level m_w describes the hop distance from w to the nearest leaf node in the final infected subtree, as $t \to \infty$.

At time $t = 0$, the source chooses a neighbor uniformly at random (e.g., node 1) and passes the message and metadata $(p_1 = 0, \ u_1 =\uparrow, \ m_1 = 1)$. Figure 13 illustrates an example spread, in which node 0 passes the message to node

1. Yellow denotes *spine* nodes, which receive the message with $u_w = \uparrow$, and gray denotes those that receive it with $u_w = \downarrow$. Whenever a node w receives a message, there are two cases. If $u_w = \uparrow$, node w chooses another neighbor z uniformly at random and forwards the message with 'up' metadata: $(p_z = w,\ u_z = \uparrow,\ m_z = m_w + 1)$. All of w's remaining neighbors z' receive the message with 'down' metadata: $(p_{z'} = w,\ u_{z'} = \downarrow,\ m_{z'} = m_w - 1)$. For instance, in Figure 13, node 1 passes the 'up' message to node 2 and the 'down' message to node 3. On the other hand, if $u_w = \downarrow$ *and* $m_w > 0$, node w forwards the message to all its remaining neighbors with 'down' metadata: $(p_z = w,\ u_z = \downarrow,\ m_z = m_w - 1)$. If a node receives $m_w = 0$, it does not forward the message further. Algorithm 3 describes this process more precisely.

Observe that adaptive diffusion ensures that the infected subgraph is a balanced tree with the true source at one of the leaves. Moreover, unlike regular diffusion, the message does not reach all the nodes in the network under adaptive diffusion (even when $T = \infty$). Even though this may seem like a fundamental drawback for adaptive diffusion, it can be shown that the infected subgraph has a size proportional to $(d-1)^{T/2}$ on regular trees (compared to $(d-1)^T$ under regular diffusion). More critically, real social networks have cycles, so neighbors of nodes with $m_w = 0$ can still get the message from other nodes in the network [14].

As before, this protocol ensures that the infected subgraph is a symmetric tree with the true source at one of the leaves. The key difference between Protocol 1 (naive adaptive diffusion) with $\alpha_d(t, h) = 0$ and Protocol 3 (Tree Protocol) is that the latter does not rely on message-passing from the virtual source to control spreading. Instead, it passes enough control information to realize the same spreading pattern in a fully-distributed fashion.

Protocol 3 Tree Protocol

```
Input: contact network $G = (V, E)$, source $v^*$, time $T$
Output: infected subgraph $G_T = (V_T, E_T)$
$V_0 \leftarrow \{v^*\}$
$m_{v^*} \leftarrow 0$ and $u_{v^*} \leftarrow \uparrow$
$v^*$ selects one of its neighbors $w$ at random
$V_1 \leftarrow V_0 \cup \{w\}$
$m_w \leftarrow 1$ and $u_w \leftarrow \uparrow$
$t \leftarrow 2$
for $t \leq T$ do
    for all $v \in V_{t-1}$ with uninfected neighbors and $m_v > 0$ do
        if $u_v = \uparrow$ then
            $v$ selects one of its uninfected neighbors $w$ at random
            $V_t \leftarrow V_{t-1} \cup \{w\}$
            $m_w \leftarrow m_w + 1$ and $u_w \leftarrow \uparrow$
        for all uninfected neighboring nodes $z$ of $v$ do
            $V_t \leftarrow V_{t-1} \cup \{z\}$
            $u_z \leftarrow \downarrow$ and $m_z \leftarrow m_v - 1$
    $t \leftarrow t + 1$
```

In the spy-based adversarial model, each spy s_i in the network observes any received messages, the associated metadata, and a timestamp T_{s_i}. Figure 14 illustrates the information observed by each spy node, where spies are outlined in red.

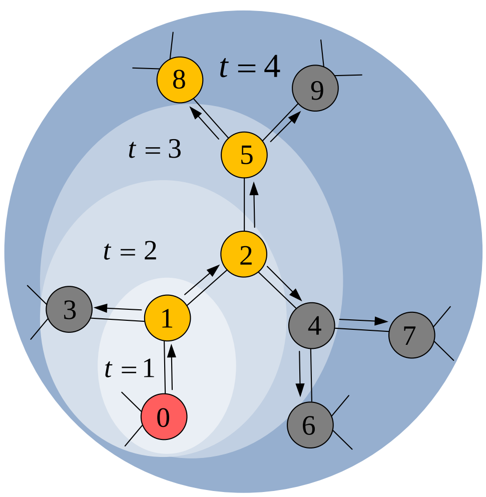


Fig. 13: Message spread using the tree protocol from [14].

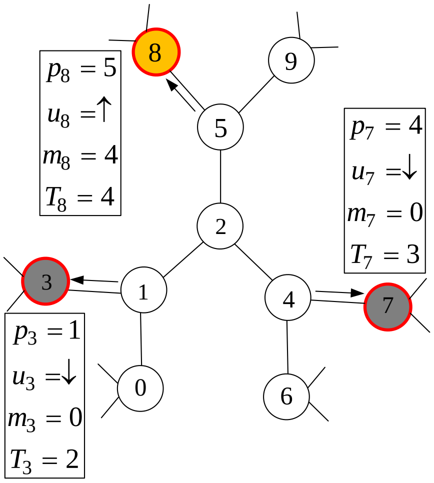


Fig. 14: The information observed by the spy nodes 3, 7, and 8 for the spread in Figure 13. Timestamps in this figure are absolute, but they need not be.

Source Estimation. The ML source-estimation algorithm for this spreading and adversarial model is described in [14]. The ML estimation algorithm is not necessary to understand this paper's primary contributions. We include it in this section for completeness, and because the probability of detection for the spy+snapshot adversarial model in Section VI uses terminology that is introduced in this estimator.

To a snapshot adversary, all leaves in the infected subgraph have the same likelihood. Because adaptive diffusion has deterministic timing, spies only help the estimator discard candidate nodes. We assume the message spreads for infinite time. There is at least one spy on the spine; consider the first such spy to receive the message, s_0. This spine spy (along with its parent and level metadata) allows the estimator to specify a *feasible subtree* in which the true source must lie. In

Figure 13, node 8 is on the spine with level $m_8 = 4$, so the feasible subtree is rooted at node 5 and contains all the pictured nodes except node 8 (9's children and grandchildren also belong, but are not pictured). Spies outside the feasible subtree do not influence the estimator, because their information is independent of the source conditioned on s_0's metadata. Only leaves of the feasible subtree could have been the source—e.g., nodes 0, 3, 6, and 7, as well as 9's grandchildren.

Protocol 4 ML Source Estimator for Algorithm 3

```
Input: contact network $G = (V, E)$, spy nodes $S = \{s_0, s_1 \dots\}$ and metadata $s_i : (p_{s_i}, m_{s_i}, u_{s_i})$
Output: ML source estimate $\hat{v}_{\mathrm{ML}}$
Let $s_0$ denote the lowest-level spine spy, with metadata $(p_{s_0}, m_{s_0}, u_{s_0})$.
$\tilde{V} \leftarrow \{v \in V :\ \delta_H(v, s_0) \le m_{s_0}$ and $p_{s_0} \in \mathcal{P}(v, s_0)\}$
$\tilde{E} \leftarrow \{(u, v) : (u, v) \in E$ and $u, v \in \tilde{V}\}$
Define the feasible subgraph as $F(\tilde{V}, \tilde{E})$
$L \leftarrow \emptyset$                                   ▷ Set of feasible pivots
$K \leftarrow \emptyset$                                   ▷ Set of eliminated pivot neighbors
for all $s \in S$ with $s \in \tilde{V}$ do
    Let $\begin{bmatrix} h_{s,\ell_s} \\ h_{\ell_s,s_0} \end{bmatrix} = \frac{1}{2}\begin{bmatrix} 1 & -1 \\ 1 & 1 \end{bmatrix} \cdot \begin{bmatrix} |P(s, s_0)| \\ T_{s_0} - T_s \end{bmatrix}$
    $\ell_s \leftarrow v \in \mathcal{P}(s, s_0) :\ \delta_H(s, \ell_s) = h_{s,\ell_s}$
    $k_s \leftarrow v \in \mathcal{P}(s, s_0) :\ \delta_H(s, k_s) = h_{s,\ell_s} - 1$
    $L \leftarrow L \cup \{\ell_s\}$                       ▷ Add pivot
    $K \leftarrow K \cup \{k_s\}$                          ▷ Add pivot neighbor
Find the lowest-level pivot: $\ell_{min} \leftarrow \mathrm{argmin}_{\ell \in L} m_\ell$
$U \leftarrow \emptyset$                                   ▷ Candidate sources
for all $v \in \tilde{V}$ where $v$ is a leaf in $F(\tilde{V}, \tilde{E})$ do
    if $\mathcal{P}(v, \ell_{min}) \cap K = \emptyset$ then
        $U \leftarrow U \cup \{v\}$
return $\hat{v}_{\mathrm{ML}}$, drawn uniformly from $U$
```

The estimator then uses spies *within* the feasible subtree to prune out candidates. The goal is to identify nodes in the feasible subtree that are on the spine and close to the source. For each spy in the feasible subtree, there exists a unique path to the spine spy s_0, and at least one node on that path is on the spine; the spies' metadata reveals the identity and level of the spine node on that path with the lowest level—we call this node a *pivot* (details in Algorithm 4). For instance, in Figure 14, we can use spies 7 and 8 to learn that node 2 is a pivot with level $m_2 = 2$. Estimation hinges on the minimum-level pivot across all spy nodes, ℓ_{min}. In the example, $\ell_{min} = 1$, since spies 3 and 8 identify node 1 as a pivot with level $m_1 = 1$. The true source must lie in a subtree rooted at a neighbor of ℓ_{min}, with no spies. In our example, this leaves only node 0, the true source.

Anonymity properties. This ML estimation procedure can be analyzed to exactly compute the probability of detection for adaptive diffusion on a d-regular tree:

Theorem 5.3 ([14]): Suppose the contact network is a regular tree with degree $d > 2$. There is a source node v^*, and each node other than the source is chosen to be a spy node i.i.d. with probability p as described in the spy model. Against colluding spies attempting to detect the location of the source, adaptive diffusion achieves the following:

(a) The probability of detection is

$$\mathbb{P}(\hat{v}_{\mathrm{ML}} = v^*) = p + \frac{1}{d-2} - \sum_{k=1}^{\infty} \frac{q_k}{(d-1)^k} \ ,$$

where $q_k \equiv (1 - (1 - p)^{((d-1)^k - 1)/(d-2)})^{d-1} + (1 - p)^{((d-1)^{k+1} - 1)/(d-2)}$.

(b) The expected distance between the source and the estimate is bounded by

$$\mathbb{E}[\delta_H(\hat{v}_{\mathrm{ML}}, v^*)] \ge 2 \sum_{k=1}^{\infty} k \cdot r_k$$

where $|T_{d,k}| = \frac{(d-1)^k - 1}{d-2}$, and $r_k \equiv \frac{1}{d-1}\Big((1 - (1 - p)^{|T_{d,k}|})^{d-1} + (d-1)(1-p)^{|T_{d,k}|} - (d-2)(1-p)^{|T_{d,k}|(d-1)} - 1\Big)$.

There are two main observations to note regarding this result:

(1) Asymptotically optimal probability of detection: As tree degree d increases, the probability of detection converges to the degree-independent fundamental limit in Proposition 5.1, i.e., $\mathbb{P}(V^* = \hat{v}_{\mathrm{ML}}) = p$. This is in contrast to diffusion, whose probability of detection tends to 1 asymptotically in d.

(2) Expected hop distance asymptotically increasing: We observe empirically that for regular diffusion, $\mathbb{E}[\delta_H(\hat{v}_{\mathrm{ML}}, v^*)]$ approaches 0 as d increases. On the other hand, for adaptive diffusion with a fixed $p > 0$, as $d \to \infty$, $\limsup \mathbb{E}[\delta_H(\hat{v}_{\mathrm{ML}}, v^*)] = 2(1 - p)$.

These observations suggest that adaptive diffusion exhibits provably stronger anonymity properties than standard diffusion on regular trees—a suggestion that is backed up by simulations on irregular trees and the Facebook graph in [14].

VI. SPY+SNAPSHOT ADVERSARIAL MODEL

The spy+snapshot adversarial model considers a natural combination of the snapshot and spy-based adversaries. At a certain time T, the adversary collects a snapshot of the infection pattern, G_T. It also collects metadata from all spies that have seen the message up to (and including) time T. Based on these two sets of metadata, the adversary infers the source.

Notably, this stronger model does not significantly impact the probability of detection as time increases. The snapshot helps detection when there are few spies by revealing which nodes are true leaves. This effect is most pronounced for small T and/or small p. The exact probability of detection at time

T is given below:

$$
\begin{aligned}
&\mathbb{P}(\hat{v}_{\mathrm{ML}} = v^*) \ = \\
&\underbrace{\frac{(1-p)^{|S_{d,T}|-1}}{|\partial S_{d,T}|}}_{\text{no spy}} + \sum_{k=1}^{T/2} \Big\{ \quad \underbrace{\frac{(1-p)^{(|T_{d,k}|-1)}\,p}{|\partial T_{d,k}|}}_{\ell_{min}\ (k^{th}\text{ spine node) is a spy}} \quad + \\
&\underbrace{(1-p)^{|T_{d,k}|}(1-(1-p)^{|S_{d,T}|-|T_{d,k+1}|})\mathbb{E}_X\Big[\frac{\mathbb{I}(X \neq d-2)}{(X+1)\,|\partial T_{d,k}|}\Big]}_{\ell_{min}\ (k^{th}\text{ spine node) not a spy}} + \\
&\underbrace{(1-p)^{|S_{d,T}|-(|T_{d,k+1}|-|T_{d,k}|)}\mathbb{E}_X\Big[\frac{\mathbb{I}(X \neq d-2)}{|\partial S_{d,T}|-(d-2-X)|\partial T_{d,k}|}\Big]}_{\text{all spy descendants of } k\text{-th spine node}} \Big\}\,, \qquad (17)
\end{aligned}
$$

where $X \sim \mathrm{Binom}(d-2, (1-p)^{|T_{d,k}|})$, $|T_{d,k}| = \frac{(d-1)^k-1}{d-2}$ is the number of nodes in each candidate subtree for a pivot at level k, and $|\partial T_{d,k}| = (d-1)^{k-1}$ is the number of leaf nodes in each candidate subtree. This expression can be evaluated numerically, as shown in Figure 15, which illustrates the tradeoff between the effect of a snapshot and spy nodes. The derivation for this expression is included in [14].

VII. Connections to Pólya's urn processes

In this section, we make a connection between adaptive diffusion on a line and Pólya's urn processes. In doing so, we highlight a property of Pólya's urn processes, which inherently provides privacy. Further, we apply the Bayesian interpretation of Pólya's urn processes to design a new implementation of adaptive diffusion and analyze the precise cost of revealing the control packets to the spy nodes, in terms of leaked anonymity.

To separately characterize the price of timestamp metadata and control packets, we focus on the concrete example of a line graph. Consider a line graph in which nodes 0 and $n+1$ are spies. One of the n nodes between the spies is chosen uniformly at random as a source, denoted by $v^* \in \{1, \dots, n\}$. We let t_0 denote the time the source starts propagating the message according to some global reference clock. Let $T_{s_1} = T_1 + t_0$ and $T_{s_2} = T_2 + t_0$ denote the timestamps when the two spy nodes receive the message, respectively. Knowing the spreading protocol and the metadata, the adversary uses the maximum likelihood estimator to optimally estimate the source.

Standard diffusion. Consider a standard discrete-time random diffusion with a parameter $q \in (0,1)$ where each uninfected neighbor is infected with probability q. The adversary observes T_{s_1} and T_{s_2}. Knowing the value of q, it computes the ML estimate $\hat{v}_{\mathrm{ML}} = \arg\max_{v \in [n]} \mathbb{P}_{T_1 - T_2 | V^*}(T_{s_1} - T_{s_2} | v)$, which is optimal assuming a uniform prior on v^*. Since t_0 is not known, the adversary can only use the difference $T_{s_1} - T_{s_2} = T_1 - T_2$ to estimate the source. We can exactly compute the corresponding probability of detection; Figure 16 (bottom panel) illustrates that the posterior (and the likelihood) is concentrated around the ML estimate, and the source can only hide among $O(\sqrt{n})$ nodes. The detection probability correspondingly scales as $1/\sqrt{n}$ (top panel).

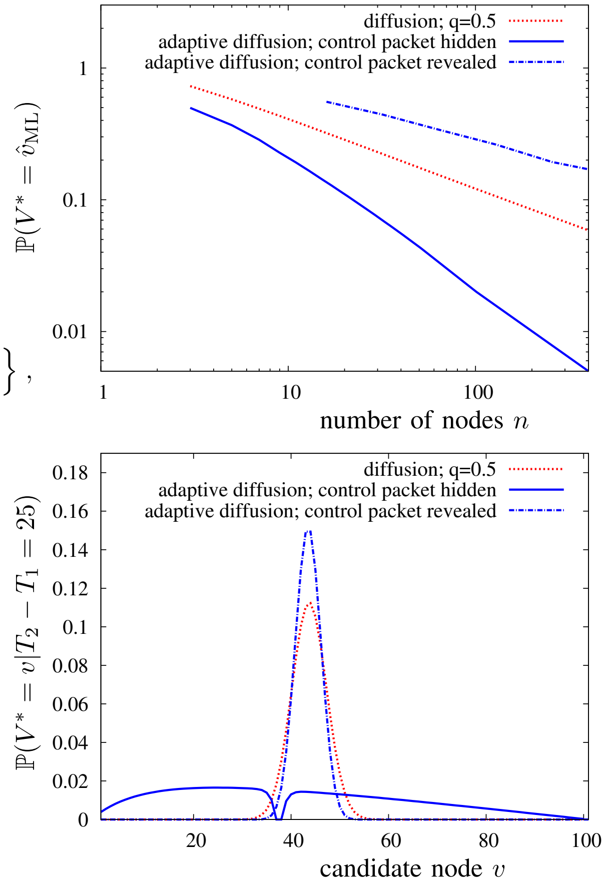


Fig. 16: Comparisons of probability of detection as a function of n (top) and the posterior distribution of the source for an example with $n = 101$ and $T_2 - T_1 = 25$ (bottom). The line with 'control packet revealed' uses the Pólya's urn implementation.

Adaptive diffusion on a line. First, recall the adaptive diffusion (Protocol 1) with the choice of $\alpha_d(h,t) = \frac{t-2h+2}{t+2}$ (Equation (2)) on a line illustrated in Figure 17. At $t=0$, the message starts at node 0. The source passes the virtual source to node 1, so $v_2 = 1$. The next two timesteps ($t = 1, 2$) are used to restore symmetry about v_2. At $t = 2$, the virtual source stays with probability $\alpha_2(2,1) = 1/2$. Since the virtual source remained fixed at $t = 2$, at $t = 4$ the virtual source stays with probability $\alpha_2(4,1) = 2/3$. The key property is that if the virtual source chooses to remain fixed at the beginning of this random process, it is more likely to remain fixed in the future, and vice versa. This is closely related to the well-known concept of *Pòlya's urn processes*; we make this connection more precise later in this section.

The protocol keeps the current virtual source with probability $\frac{2\delta_H(v_t,v^*)}{t+2}$, where $\delta_H(v_t, v^*)$ denotes the hop distance between the source and the virtual source, and passes it otherwise. The control packet therefore contains two pieces of information: $\delta_H(v_t, v^*)$ and t.

Suppose spy nodes only observed timestamps and parent nodes but *not* control packets. The adversary could then numerically compute the ML estimate $\hat{v}_{\mathrm{ML}} =$

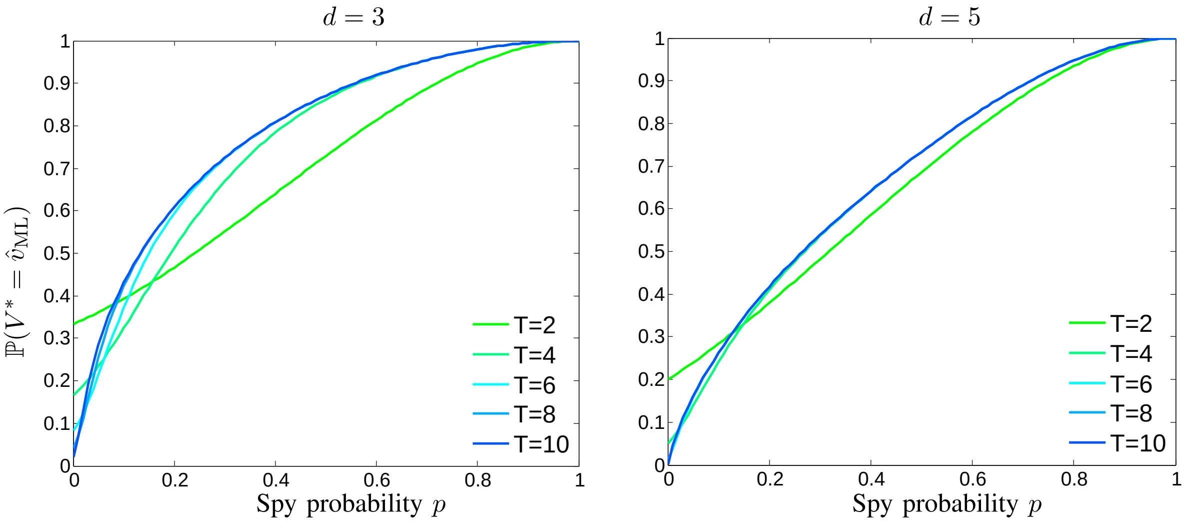


Fig. 15: Probability of detection under the spy+snapshot adversarial model. As estimation time and tree degree increase, the effect of the snapshot on detection probability vanishes.

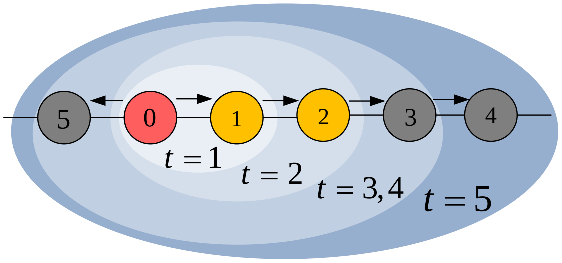


Fig. 17: Spreading on a line. The red node is the message source. Yellow nodes denote nodes that have been, are, or will be the center of the infected subtree.

$\arg\max_{v\in[n]} \mathbb{P}_{T_1-T_2|V^*}(T_{s_1} - T_{s_2}|v)$. We can compute the corresponding detection probability exactly. Figure 16 shows the posterior is close to uniform (top panel) and the probability detection would scale as $1/n$ (bottom panel), which is the best one can hope for. Of course, spies *do* observe control packets, so they can learn $\delta_H(v^*, v_T)$ and identify the source with probability 1. We therefore introduce a new adaptive diffusion implementation that is robust to control packet information.

Adaptive diffusion via Pólya's urn. The random process governing the virtual source's propagation under adaptive diffusion is identical to a Pólya's urn process [47]. We propose the following alternative implementation of adaptive diffusion. At $t = 0$ the protocol decides whether to pass the virtual source left ($D = \ell$) or right ($D = r$) with probability half. Let D denote this random choice. Then, a latent variable q is drawn from the uniform distribution over $[0, 1]$. Thereafter, at each even time t, the virtual source is passed with probability q or kept with probability $1 - q$. It follows from the Bayesian interpretation of Pólya's urn processes that this process has the same distribution as the adaptive diffusion process.

Further, in practice, the source could simulate the whole process in advance. The control packet would simply reveal to each node how long it should wait before further propagating the message. Under this implementation, spy nodes only observe timestamps T_{s_1} and T_{s_2}, parent nodes, and control packets containing the infection delay for the spy and all its descendants in the infection. Given this, the adversary can exactly determine the timing of infection with respect to the start of the infection T_1 and T_2, and also the latent variables D and q. A proof of this statement and the following proposition is provided in Section IX-G. The next proposition provides an upper bound on the detection probability for such an adversary.

Proposition 7.1: When the source is uniformly chosen from n nodes between two spy nodes, the message is spread according to adaptive diffusion, and the adversary has a full access to the time stamps, parent nodes, and the control packets that is received by the spy nodes, observations T_1, T_2, q and D, the adversary can compute the ML estimate:

$$\hat{v}_{\mathrm{ML}} = \begin{cases} \frac{T_1+2}{2} + \left\lfloor q\left(\frac{T_1-2}{2}\right)\right\rfloor & , \text{ if } T_1 \text{ even and } D = \ell\,, \\ \frac{T_1+3}{2} + \left\lfloor q\left(\frac{T_1-1}{2}\right)\right\rfloor & , \text{ if } T_1 \text{ odd and } D = \ell\,, \\ 1 + \left\lfloor (1-q)\left(\frac{T_1-1}{2}\right)\right\rfloor & , \text{ if } T_1 \text{ odd and } D = r\,. \end{cases}$$

where T_1 is the time since the start of the spread until s_1 receives the message, and q is the hidden parameter of the Pólya's urn process, and D is the initial choice of direction for the virtual source. This estimator achieves a detection probability upper bounded by

$$\mathbb{P}\big(\, V^* = \hat{v}_{\mathrm{ML}}\,\big) \le \frac{\pi\sqrt{8}}{\sqrt{n}} + \frac{2}{n}. \tag{18}$$

Equipped with an estimator, we can also simulate adaptive diffusion on a line. Figure 16 (top) illustrates that even with access to control packets, the adversary achieves probability of detection scaling as $1/\sqrt{n}$ – similar to standard diffusion. For a given value of T_1, the posterior and the likelihood are concentrated around the ML estimate, and the source can only hide among $O(\sqrt{n})$ nodes, as shown in the bottom panel for $T_1 = 58$. In the realistic adversarial setting where control packets are revealed at spy nodes, adaptive diffusion can only hide as well as standard diffusion over a line.

VIII. Future directions and connections to game theory

This work raises a number of interesting open questions. One such question is how to give plausible deniability to users that "approve" sensitive messages. Initial investigations exist [48], but technical and algorithmic questions remain, such as fundamental tradeoffs between privacy, spreading rate, and the utility lost, measured by the increased amount of spam in the network.

Another possible direction harnesses the recently-shown idea that it is difficult to locate multiple diffusion sources [49]. It might be possible to spread the messages faster than adaptive diffusion and still achieve perfect obfuscation by creating multiple pseudo-sources. This approach sacrifices some social relevance, since the message spreads from remote nodes that may not necessarily like the message.

Finally, recent work explores how to identify the first node in a randomly growing network [50], [51], [52], [53]. Unlike diffusion, the network itself is growing according to some mechanism, e.g., preferential attachment. [50], explores fundamental limits on the size of a candidate set to ensure that the true *source* or root of the random network is in the candidate set with high probability. A natural question is how to grow such a network to hide the identity of the root.

Connections to game theory. Consider a game-theoretic setting where there are two players, the protocol designer and the adversary. The designer can choose any strategy to spread the message from a source v^*, as long as the message is passed one hop at a time. The adversary can choose any strategy (computationally expensive or not) to compute an estimated source $\hat{v}$ given a some side information on the spread. As a result, the source can either be detected or not. In terms of the payoff, the protocol designer wants to minimize the probability of detection and the adversary wants to maximize it.

In this static game setting, the adaptive diffusion is a (weak) dominant strategy under a certain condition. Consider a snapshot-based adversary and a contact network of d-regular tree. The special condition we impose is that we are only allowed protocols that infect at most, say, $1 + (2(d-1)^{(T+1)/2} - d)/(d-2)$ nodes. In this setting, Theorem 4.1 implies that adaptive diffusion is dominant up to a vanishing additive factor.

Following our work [45], a game-theoretic formulation of the problem of source obfuscation was recently proposed in [54]. The designer is restricted to use deterministic protocols, and the snapshot-based adversary is restricted to use a certain family of estimators based on Jordan centers. Under these restrictions, it is shown that there is no "dominant" protocol in Nash equilibrium sense, other than the simple (deterministic) diffusion.

There are several interesting future research directions. First, when infecting more nodes is of priority, a fundamental question is whether there is a dominant strategy for a given target infection rate. Adaptive diffusion achieves the fundamental limit of $\mathbb{P}(\text{detection}) = 1/N_T$ until $N_T \leq 1 + (2(d-1)^{(T+1)/2} - d)/(d-2) \simeq (d-1)^{(T/2)}$ (see Figure 18) on d-regular trees. It is an open question what the fundamental limit is above this threshold, and if there is efficient distributed protocol achieving this optimal tradeoff. In particular, if we have to spread every time deterministically to achieve the infection speed of $N_T \simeq (d-1)^T$, then the source will be trivially detected as the center of infection. Above the threshold of $\log N_T \simeq \frac{1}{2}T\log(d-1)$, A variant of adaptive diffusion can achieve the infection rate $\alpha T \log(d-1)$ with probability of detection $(\alpha - 1)T\log(d-1)$ for any $\alpha \in [0.5, 1]$. Hence, all grey triangular region is achievable by adaptive diffusion in Figure 18.

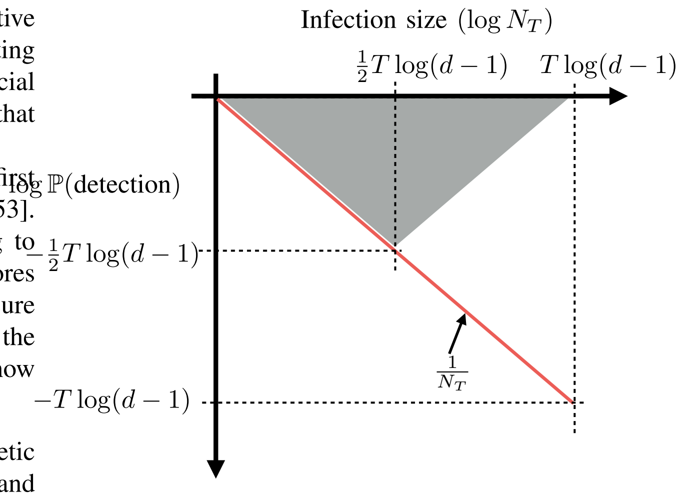


Fig. 18: The fundamental limit of $\mathbb{P}(\text{detection}) \geq 1/N_T$ is shown in a solid red line. This is achieved by adaptive diffusion until $\log(N_T) \leq \frac{1}{2}T\log(d-1)$. Infection size at time T is shown on the x-axis in log-scale and the probability of detection on y-axis also in log-scale.

Second, when the same source spreads multiple messages that can be linked, this can be posed as a dynamic game. If the adversary observes multiple spreads of infection from a single source, how much does the probability of detection increase as a function of the multiplicity of the spread? One possibility is to spread according to adaptive diffusion the first time, and use exactly the same pattern of spread in the consecutive spread of the following messages from the same source. Hence, from the meta-data, there is no more information on who the source is. However, this creates a certain permanent bias in the spread, which may be undesirable, depending on the application.

Next, a set of nodes can collude to spread the exactly same message, but starting from multiple sources simultaneously with possible delays. Unless carefully coordinated, such spread from multiple sources can be easily detected [55] and there is no gain in collusion. However, we can consider an alternative strategy of creating a pseudo-source node to make the source hard to find. At a certain time (possible $t = 0$), the protocol starts another chain of spread starting from a node far away from the infection so far. This can improve the detection probability by a factor of the number of such new infections, at a price of losing the benefits of social filtering and possibly spamming the users with irrelevant messages. We want to be able to measure such a loss in social filtering and characterize the tradeoff.

IX. Proofs

A. Proof of Theorem 4.1

Spreading rate. Under Protocol 1, G_T is a complete $(d-1)$-ary tree (with the exception that the root has d children) of depth $T/2$ whenever T is even. Whenever T is odd, with probability $\alpha_d(T,h)$, G_T is again such a $(d-1)$-ary tree of depth $(T+1)/2$. With probability $1-\alpha_d(T,h)$, G_T is made up of two $(d-1)$-ary trees of depth $(T-1)/2$ each with their roots connected by an edge. Therefore, it follows that when $d>2$, N_T is given by

$$N_T \ = \ \begin{cases} 1, & T=0, \\ \frac{2(d-1)^{(T+1)/2}}{d-2} - \frac{2}{d-2}, & T\geq 1,\ T \text{ odd, w.p. } (1-\alpha)\ , \\ \frac{d(d-1)^{(T+1)/2}}{d-2} - \frac{2}{d-2}, & T\geq 1,\ T \text{ odd, w.p. } \alpha\ , \\ \frac{d(d-1)^{T/2}}{d-2} - \frac{2}{d-2}, & T\geq 2,\ T \text{ even }; \end{cases}$$

Similarly, when $d=2$, N_T can be expressed as follows:

$$N_T \ = \ \begin{cases} 1, & T=0, \\ T+1, & T\geq 1,\ T \text{ odd, w.p. } (1-\alpha)\ , \\ T+2, & T\geq 1,\ T \text{ odd, w.p. } \alpha\ , \\ T+2, & T\geq 2,\ T \text{ even }; \end{cases}$$

The lower bound on N_T in equation (3) follows immediately from the above expressions.

Probability of detection. For any given infected graph G_T, the virtual source v_T cannot have been the source node, since the true source always passes the token at timestep $t=1$. So $\mathbb{P}(G_T|v=v_T)=0$. We claim that for any two nodes that are not the virtual source at time T, $u,w\in G_T$, $\mathbb{P}(G_T|u)=\mathbb{P}(G_T|w)>0$. This is true iff for any non-virtual-source node v, there exists a sequence of virtual sources $v_i{}_{i=0}^{T}$ that evolves according to Protocol 1 with $v_0=v$ that results in the observed G_T, and for all $u,w\in G_T\setminus\{v_T\}$, this sequence has the same likelihood. In a tree, a unique path exists between any pair of nodes, so we can always find a valid path of virtual sources from a candidate node $u\in G_T\setminus\{v_T\}$ to v_T. We claim that any such path leads to the formation of the observed G_T. Due to regularity of G and the symmetry in G_T, for even T, $\mathbb{P}(G_T|v^{(1)})=\mathbb{P}(G_T|v^{(2)})$ for all $v^{(1)},v^{(2)}\in G_T$ with $\delta_H(v^{(1)},v_T)=\delta_H(v^{(2)},v_T)$. Moreover, recall that the $\alpha_d(t,h)$'s were designed to satisfy the distribution in equation (1). Combining these two observations with the fact that we have $(d-1)^h$ infected nodes h-hops away from the virtual source, we get that for all $v^{(1)},v^{(2)}\in G_T\setminus\{v_T\}$, $\mathbb{P}(G_T|v^{(1)})=\mathbb{P}(G_T|v^{(2)})$. For odd T, if the virtual source remains the virtual source, then G_T stays symmetric about v_T, in which case the same result holds. If the virtual source passes the token, then G_T is perfectly symmetric about the edge connecting v_{T-1} and v_T. Since both nodes are virtual sources (former and present, respectively) and $T>1$, the adversary can infer that neither node was the true source. Since the two connected subtrees are symmetric and each node within a subtree has the same likelihood of being the source by construction (equation (1)), we get that for all $v^{(1)},v^{(2)}\in G_T\backslash\{v_T,v_{T-1}\}$, $\mathbb{P}(G_T|v^{(1)})=\mathbb{P}(G_T|v^{(2)})$. Thus at odd timesteps, $\mathbb{P}(\hat{v}_{ML}=v^*)\geq 1/(N_T-2)$.

B. Proof of Proposition 4.2

First, under Protocol 1 (adaptive diffusion) with $\alpha_d(t,h)=0$, G_T is a complete $(d-1)$-ary tree (with the exception that the root has d children) of depth $T/2$ whenever T is even. G_T is made up of two complete $(d-1)$-ary trees of depth $(T-1)/2$ each with their roots connected by an edge whenever T is odd. Therefore, it follows that N_T is a deterministic function of T and is given by

$$N_T \ = \ \begin{cases} 1, & T=0, \\ \frac{2(d-1)^{(T+1)/2}}{d-2} - \frac{2}{d-2}, & T\geq 1,\ T \text{ odd }, \\ \frac{d(d-1)^{T/2}}{d-2} - \frac{2}{d-2}, & T\geq 2,\ T \text{ even }; \end{cases}$$

The lower bound on N_T in equation (5) follows immediately from the above expression.

For any given infected graph G_T, it can be verified that any non-leaf node could not have generated G_T under the Tree Protocol. In other words, $\mathbb{P}(G_T|v \text{ non-leaf node})=0$ and v could not have started the rumor. On the other hand, we claim that for any two leaf nodes $v_1,v_2\in G_T$, we have that $\mathbb{P}(G_T|v_1)=\mathbb{P}(G_T|v_2)>0$. This is true because for each leaf node $v\in G_T$, there exists a sequence of state values $\{s_{1,u},s_{2,u}\}_{u\in G_T}$ that evolves according to the Tree Protocol with $s_{1,v}=1$ and $s_{2,v}=0$. Further, the regularity of the underlying graph G ensures that all these sequences are equally likely. Therefore, the probability of correct rumor source detection under the maximum likelihood algorithm is given by $\mathbb{P}_{ML}(T)=1/N_{l,T}$, where $N_{l,T}$ represents the number of leaf nodes in G_T. It can be also shown that $N_{l,T}$ and N_T are related to each other by the following expression

$$N_{l,T}=\frac{(d-2)N_T+2}{d-1}.$$

This proves the expression for $\mathbb{P}\big(\hat{v}_{\rm ML}=v^*\big)$ given in equation (6).

Expected distance. For any $v^*\in G$ and any T, $\mathbb{E}[\delta_H(v^*,\hat{v}_{\rm ML})]$ is given by

$$\mathbb{E}[\delta_H(v^*,\hat{v}_{\rm ML})]=\sum_{v\in G}\sum_{G_T}\mathbb{P}(G_T|v^*)\mathbb{P}(\hat{v}_{\rm ML}=v)\delta_H(v^*,v).$$

As indicated above, no matter where the rumor starts from, G_T is a $(d-1)$-ary tree (with the exception that the root has d children) of depth $T/2$ whenever T is even. Moreover, $\hat{v}_{\rm ML}=v$ with probability $1/N_{l,T}$ for all v leaf nodes in G_T. Therefore, the above equation can be solved exactly to obtain the expression provided in the statement of the proposition.

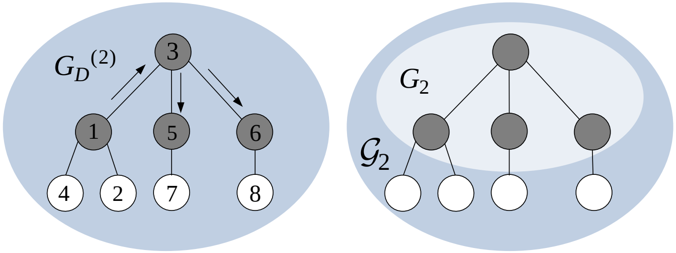


Fig. 19: One realization of the random, irregular-tree branching process. Although each realization of the random process $G_D^{(t)}$ yields a labelled graph, the adversary observes G_T and $\mathcal{G}_T$, which are *unlabelled*. White nodes are uninfected, grey nodes are infected.

C. *Proof of Proposition 4.3*

We upper bound the probability of detection by assuming that the adversary takes a snapshot at *every* time step after T; the adversary can also learn the exact value of T by noting the size of the snapshots in successive time steps. The structure of all snapshots after G_T depends deterministically on the binary timeseries of choices to either keep the virtual source token, or to pass it, in each time step after T—we refer to this timeseries as $\mathcal{K}_T$. The timeseries $\mathcal{K}_T$, in turn, is random, with values that depend probabilistically on only the timestamp (which is known to the adversary), the tree degree (known), and the virtual source's distance from the true source (unknown). Because adaptive diffusion does not allow the virtual source to "backtrack", or move closer to the true source over time, the (unique) path from the true source to the virtual source v_T at time T cannot intersect the path comprised of the virtual sources after time T—call it $\mathcal{P}_T$—except possibly at v_T itself. Therefore, let us consider the first node in $\mathcal{P}_T$ that is not equal to v_T; we call it $v_{T'}$. $v_{T'}$ is necessarily a neighbor of v_T. Then let us define the largest possible subtree of G_T that is rooted at $v_{T'}$ and does *not* contain v_T; we call this subtree $\mathcal{T}_T$.

Now, suppose that by observing the timeseries $\mathcal{K}_T$, the adversary could learn the distance between v^* and v_T exactly (this is a worst-case assumption). Let us call that distance L. Then the source is equally likely to be any node w at a distance of L hops from v_T, such that $w \notin \mathcal{T}_T$. Therefore, we can upper bound the probability of detection by conditioning on L, and counting the number of feasible nodes w.

We assume for the sake of simplicity that all snapshots are taken at even time steps (including G_T), since snapshots at odd time steps do not contribute any additional information, i.e., if the adversary observed G_T at an odd timestep, it could recover G_{T-1} from the subsequent observed snapshots, which is equivalent to observing the first snapshot at time $T-1$. Then

$$\mathbb{P}(\hat{v}_{ML} = v^*) = \sum_{\ell=1}^{T/2} \mathbb{P}(L=\ell)\mathbb{P}(\hat{v}_{ML} = v^* | L = \ell)$$

From the previous argument, we have

$$\mathbb{P}(\hat{v}_{ML} = v^* | L = \ell) = \frac{1}{(d-1)^\ell},$$

instead of $1/d(d-1)^{\ell-1}$, since the entire subtree of G_T containing $\mathcal{P}_T$ is excluded from the set of possible candidate sources. Additionally, it is straightforward to compute $\mathbb{P}(L=\ell)$ from the properties of adaptive diffusion:

$$\mathbb{P}(L=\ell) = \frac{(d-2)(d-1)^{\ell-1}}{(d-1)^{T/2}-1},$$

so the overall probability of detection is

$$\mathbb{P}(\hat{v}_{ML} = v^*) \quad = \quad \frac{d-2}{d-1} \cdot \frac{1}{(d-1)^{T/2}-1} \cdot \frac{T}{2}.$$

Note that

$$N_T \quad = \quad \frac{d(d-1)^{T/2}-2}{d-2} = \frac{(d-1)^{T/2+1} + (d-1)^{T/2} - 2}{d-2}.$$

Since $\left((d-1)^{T/2+1} - 2\right) \geq (d-1)^{T/2}$ for all $d > 2$ and all even $T \geq 2$, it holds that $N_T \geq \frac{2(d-1)^{T/2}}{d-2}$. From this, we can conclude that $T/2 \leq \log_{d-1} N_T + \log_{d-1}(d/2-1)$. It also holds that for all $d > 2$, $(d-1)^{T/2} - 1 \geq \frac{N_T - 1}{3}$, so we have

$$\mathbb{P}(\hat{v}_{ML} = v^*) \leq \quad \frac{d-2}{d-1} \cdot \frac{3}{N_T - 1} \left(\log N_T + \log(d/2-1)\right),$$

which gives the claim.

D. *Proof of Theorem 4.5*

We first analyze the probability of detection for any given estimator (see equation (**??**)); we then show that the estimator in equation (11) is a MAP estimator, maximizing this probability of detection. Finally, we show that using the MAP estimator in equation (11) gives the probability of detection in equation (9).

We begin with some definitions. Consider the following random process, in which we fix a source v^* and generate a (random) labelled tree $G_D^{(t)}$ for each time t and for a given degree distribution D. At time $t = 0$, $G_D^{(t)}$ consists of a single node v^*, which is given a label 1. The source v^* draws a degree d_1 from D, and generates d_1 child nodes, labelled in order of creation (i.e., 2 through $d_1 + 1$). At the next time step, $t = 1$, the source picks one of these neighbors uniformly at random to be the new virtual source and infects that neighbor. According to Protocol 1, each time a node v is infected, v draws its degree d_v from D, then generates $d_v - 1$ labelled child nodes. So at the end of time $t = 1$, $G_D^{(1)}$ contains the source and its uninfected neighbors, as well as the new virtual source and its uninfected neighbors. An example of $G_D^{(2)}$ is shown in Figure 19 (left panel) with $d_1 = 3$ and virtual source at node 3. Grey nodes are infected and white nodes are uninfected neighbors. Note that the node labelled 1 is always exactly one hop from a leaf of $G_D^{(t)}$ for all $t > 0$; also, nodes infect their neighbors in ascending order of their labels. The leaves of $G_D^{(t)}$ represent the uninfected neighbors of infected leaves in standard adaptive diffusion spreading over a given graph. Define $\Omega_{(t,D)}$ as the set of all labelled trees generated at time t according to this random process.

At some time T, the adversary observes the snapshot of infected subgraph G_T. Notice that we do not need to generate the entire contact network, since G_T is conditionally independent of the rest of the contact network given its one-hop

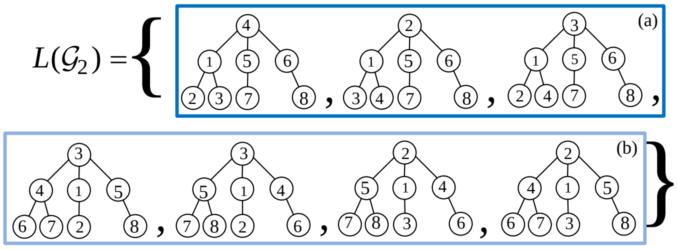


Fig. 20: $L(\mathcal{G}_2)$ for the snapshot $\mathcal{G}_2$ illustrated in Figure 19. Boxes (a) and (b) illustrate the two families partitioning $L(\mathcal{G}_2)$.

neighbors. Hence, the we only need to generate (and consider) the one hop neighbors of G_T at any given T. We use $\mathcal{G}_T$ to denote this random graph that includes G_T and its one hop neighbors as generated according to the previously explained random process. Notice that the adversary only observes $\mathcal{G}$, which is an *unlabelled* snapshot of the infection and its one hop neighbors (see Figure 19, right panel). We refer to the leaves of G_T as 'infected leaves', denoted by ∂G_T, and the leaves of $\mathcal{G}_T$ as 'uninfected leaves' denoted by $\partial \mathcal{G}_T$. Define

$$L(\mathcal{G}_T) \equiv \{\tilde{G} \in \Omega_{(T,D)} \mid U(\tilde{G}) = \mathcal{G}_T\},$$

i.e., the set of all labelled graphs (generated according to the described random process) whose unlabelled representation $U(\tilde{G})$ is equal to the snapshot $\mathcal{G}_T$. Figure 20 illustrates $L(\mathcal{G}_T)$ for the graph $\mathcal{G}_2$ in Figure 19.

We define a *family* $C_{\mathcal{G}_T,v} \subseteq L(\mathcal{G}_T)$ as the set of all labelled graphs whose labeling could have been generated by breadth-first labeling of $\mathcal{G}_T$ starting at node $v \in \partial G_T$. Here breadth-first labeling is a valid order of traversal for a breadth-first search of $\mathcal{G}_T$ starting at node v. We restrict v to be a valid source for an adaptive diffusion spread—that is, it is an infected leaf in ∂G_T. Note that a BFS labeling starting from two different nodes on the unlabelled tree can yield the same labelled graph. In Figure 20, boxes (a) and (b) illustrate the two families contained in $L(\mathcal{G}_2)$.

Let $\mathbb{P}(C_{\mathcal{G},v}) \equiv \mathbb{P}(G_D^{(T)} \in C_{\mathcal{G},v})$ denote the probability that the labelled graph $G_D^{(T)}$ whose snapshot is $\mathcal{G}$ is generated from a node v. From the definition of the random process for generating labelled graphs, we get

$$\mathbb{P}(C_{\mathcal{G}_T,v}) = \underbrace{\left(\prod_{w \in G_T} \mathbb{P}_D(d_w)\right) \underbrace{Q(\mathcal{G}_T, v)}_{\text{virtual sources}} \underbrace{|C_{\mathcal{G}_T,v}|}_{\substack{\text{count of}\\ \text{isomorphisms}}}}_{\text{degrees of } G} \qquad (19)$$

where $\mathbb{P}_D(d)$ is the probability of observing degree d under degree distribution D, and

$$Q(\mathcal{G}_T, v) = \frac{\mathbb{1}_{v \in \partial G_T}}{d_v \prod_{w \in \Phi_{v,v_T} \setminus \{v, v_T\}} (d_w - 1)}$$

is the probability of passing the virtual source from v to the virtual source v_T given the structure of $\mathcal{G}_T$, where Φ_{v,v_T} is the unique path from v to v_T in $\mathcal{G}_T$. Equation (19) holds because for all instances in $C_{\mathcal{G}_T,v}$, the probability of the degrees of the nodes and the probability of the path of the virtual source remain the same.

The probability of observing a given snapshot $\mathcal{G}_T$ is precisely $\mathbb{P}(G_D^{(T)} \in L(\mathcal{G}_T))$. Notice that $C_{\mathcal{G}_T,v}$ partitions $L(\mathcal{G}_T)$ in to family of labelled trees that are generated from the same source. This gives the following decomposition:

$$\mathbb{P}(G_D^{(T)} \in L(G_T)) = \sum_{v \in \mathcal{C}_{\mathcal{G}_T}} \mathbb{P}(C_{\mathcal{G}_T,v}), \qquad (20)$$

where we define $\mathcal{C}_{\mathcal{G}_T}$ as the set of possible candidates of the source that generate distinct labelled trees, i.e.,

$$\mathcal{C}_{\mathcal{G}_T} \equiv \{v \in G_T \mid C_{\mathcal{G}_T,v} \neq C_{\mathcal{G}_T,v'} \ \forall \ v' \in \mathcal{C}_{\mathcal{G}_T}, \ v' \neq v\} .$$

Notice that this set is not unique, since there can be multiple nodes that represent the same family $C_{\mathcal{G}_T,v}$. We pick one of such node v to represent the class of nodes that can generate the same family of labelled trees. We use this v to index these families and not to denote any particular node in ∂G_T.

Consider an estimate of the source $\hat{v}(\mathcal{G}_T)$. In general, $\hat{v}(\mathcal{G}_T)$ is a random variable, potentially selected from a set of candidates. We define detection $(\overline{D})$ as the event in which $\hat{v}(\mathcal{G}_T) = v_1(G_D^{(T)})$; i.e., the estimator outputs the node that started the random process. We can partition the set of candidate nodes ∂G_T, by grouping together those nodes that are indistinguishable to the estimator into *classes*. Precisely, we define a subset of nodes indexed by $v \in \mathcal{C}_{\mathcal{G}_T}$,

$$\chi_{\mathcal{G}_T,v} \equiv \{v' \in \partial G_T \mid C_{\mathcal{G}_T,v} = C_{\mathcal{G}_T,v'}\} .$$

For a given snapshot, there are as many classes as there are families. In Figure 20, the class associated with family (a) has one element—namely, the node labeled '1' in family (a). The class associated with family (b) contains two nodes: the node labeled '1' in family (b), and the node labeled '5' in the rightmost graph of family (b), since both nodes give rise to the same family.

We consider, without loss of generality, an estimator that selects a node in a given class with probability $\mathbb{P}(\hat{v}(\mathcal{G}_T) \in \chi_{\mathcal{G}_T,v})$. Notice that $|\chi_{\mathcal{G}_T,v}|$ denotes the number of (indistinguishable) source candidates in this class. From equation (20), the probability of detection given a snapshot is

$$\begin{aligned} \mathbb{P}(\overline{D}|\mathcal{G}_T) &= \frac{\mathbb{P}\left(G_D^{(T)} \in L(\mathcal{G}_T) \wedge \overline{D}\right)}{\mathbb{P}(G_D^{(T)} \in L(\mathcal{G}_T))}. \\ &= \frac{\sum_{v \in \mathcal{C}_{\mathcal{G}_T}} \mathbb{P}(C_{\mathcal{G}_T,v}) \mathbb{P}\left(\overline{D} \,\middle|\, G_D^{(T)} \in C_{\mathcal{G}_T,v}\right)}{\sum_{v \in \mathcal{C}_{\mathcal{G}_T}} \mathbb{P}(C_{\mathcal{G}_T,v})} \end{aligned} \qquad (21)$$

where $\mathbb{P}(\overline{D}|G_D^{(T)} \in C_{\mathcal{G}_T,v}) = \mathbb{P}(\hat{v}(\mathcal{G}_T) \in \chi_{\mathcal{G}_T,v})/|\chi_{\mathcal{G}_T,v}|$. We use the following observation:

Lemma 9.1:

$$\frac{\mathbb{P}(C_{\mathcal{G}_T,v})/|\chi_{\mathcal{G}_T,v}|}{\sum_{v \in \mathcal{C}_{\mathcal{G}_T}} \mathbb{P}(C_{\mathcal{G}_T,v})} = \frac{1}{d_{v_T} \prod_{\substack{w \in \phi(v,v_T) \\ \setminus \{v,v_T\}}} (d_w - 1)}. \qquad (22)$$

(Proof in Section IX-D1)

Substituting equation (22) into equation (21), we get that

$$\mathbb{P}(\overline{D}|\mathcal{G}_T) = \sum_{v \in \mathcal{C}_{\mathcal{G}_T}} \frac{\mathbb{P}(\hat{v}(\mathcal{G}_T) \in \chi_{\mathcal{G}_T,v})}{d_{v_T} \prod_{\substack{w \in \phi(v,v_T) \setminus \\ \{v,v_T\}}} (d_w - 1)}.$$

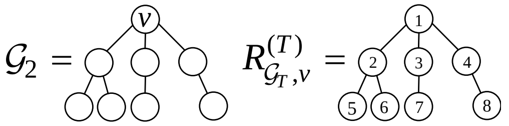


Fig. 21: A realization of the random labeling process given an unlabeled snapshot.

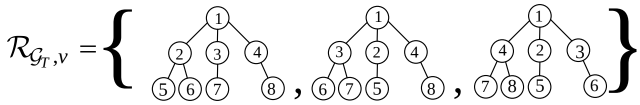


Fig. 22: The set $\mathcal{R}^{(T)}_{\mathcal{G}_T,v}$ for the snapshot and node specified in Figure 21.

Since each term of this summation is bounded by

$$\frac{\mathbb{P}(\hat{v}(\mathcal{G}_T) \in \chi_{\mathcal{G}_T,v})}{d_{v_T} \prod\limits_{\substack{w\in\phi(v,v_T)\setminus \\ \{v,v_T\}}} (d_w - 1)} \le \frac{1}{\min\limits_{v\in\mathcal{C}_{\mathcal{G}_T}} d_{v_T} \prod\limits_{\substack{w\in\phi(v,v_T) \\ \setminus\{v,v_T\}}} (d_w - 1)},$$

and $\sum_{v\in\mathcal{C}_{\mathcal{G}_T}} \mathbb{P}(\hat{v}(\mathcal{G}_T) \in \chi_{\mathcal{G}_T,v}) = 1$, it must hold that

$$\mathbb{P}(\overline{D}|\mathcal{G}_T) \quad \le \quad \frac{1}{\min\limits_{v\in\mathcal{C}_{\mathcal{G}_T}} d_{v_T} \prod\limits_{\substack{w\in\phi(v,v_T) \\ \setminus\{v,v_T\}}} (d_w - 1)}.$$

This upper bound on the detection probability is achieved exactly if we choose weight $\mathbb{P}(\hat{v}(\mathcal{G}_T) \in \chi_{\mathcal{G}_T,v}) = 1$ for the class(es) minimizing the product $\prod_{w\in\phi(v,v_T)\setminus\{v,v_T\}}(d_w - 1)$, i.e.,

$$\hat{v}(G_T) = \arg\min_{v\in\partial G_T} \prod_{\substack{w\in\phi(v,v_T) \\ \setminus\{v,v_T\}}} (d_w - 1).$$

1) Proof of Lemma 9.1: We have that

$$\mathbb{P}(C_{\mathcal{G}_T,v}) = \underbrace{\left(\prod_{w\in G_T} \mathbb{P}_D(d_w)\right)}_{\text{degrees of } G} \underbrace{Q(\mathcal{G}_T, v)}_{\text{virtual sources}} \underbrace{|C_{\mathcal{G}_T,v}|}_{\text{count of isomorphisms}}$$

where v is a feasible source for the adaptive diffusion process, i.e., a leaf of the infection G_T.

The proof of the lemma proceeds in four steps:

1) We first recursively define a function $H(\mathcal{G}_T, v)$ that is equal to $|C_{\mathcal{G}_T,v}|$. This function is defined over any balanced, undirected tree and node; the tree need not be generated via the previously-described adaptive diffusion branching process. In addition to $H(\mathcal{G}_T, v)$, we are interested in $H(\mathcal{G}_T, v_T)$.
2) We show that

$$\mathbb{P}(C_{\mathcal{G}_T,v}) \quad = \quad \left(\prod_{v\in G_T} \mathbb{P}_D(d_v)\right) H(\mathcal{G}_T, v_T) \times \frac{|\chi_{\mathcal{G}_T,v}|}{d_{v_T} \prod\limits_{\substack{w\in\phi(v,v_T) \\ \setminus\{v,v_T\}}} (d_w - 1)}.$$

3) We show that

$$\sum_{v\in\mathcal{C}_{\mathcal{G}_T,v}} \mathbb{P}(C_{\mathcal{G}_T,v}) = \left(\prod_{v\in G_T} \mathbb{P}_D(d_v)\right) H(\mathcal{G}_T, v_T). \tag{23}$$

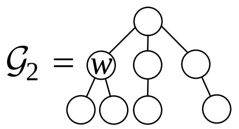
4) We combine steps (2) and (3) to show the result.

Step 1 We wish to define $H(\mathcal{G}_T, v)$—a function that counts the number of distinct, isomorphic graphs generated by a breadth-first search of a balanced tree $\mathcal{G}_T$, rooted at node v. Consider a random process defined as follows. Given $\mathcal{G}_T$ and root node v, the process starts at v and labels it 1. For each neighbor w of node 1, the process randomly orders w's unlabelled neighbors, and labels them in order of traversal. The process proceeds to label nodes in a breadth-first fashion, traversing each node's unlabelled neighbors in a randomly-selected order, until all nodes have been visited. Let $R^{(t)}_{\mathcal{G}_T,v}$ denote a labelled tree generated according to the described random process (see Figure 21).

The function $H(\mathcal{G}_T, v)$ counts the number of distinct graphs that can result from this random process over $\mathcal{G}_T$ when starting from node v. More precisely, define $\mathcal{R}^{(T)}_{\mathcal{G}_T,v}$ as the set of all possible trees $R^{(T)}_{\mathcal{G}_T,v}$ generated according to this random labeling. $H(\mathcal{G}_T, v)$ is defined as the size of $\mathcal{R}^{(T)}_{\mathcal{G}_T,v}$. Figure 22 illustrates $\mathcal{R}^{(T)}_{\mathcal{G}_T,v}$ for $\mathcal{G}_T$ and v shown in Figure 21. In that example, $H(\mathcal{G}_T, v) = 3$.

Recall that $\mathcal{G}_T$ is a balanced tree. The Jordan center of this tree is denoted by v_T. If $\mathcal{G}_T$ was generated according to adaptive diffusion, v_T would be the virtual source at time T. Although we say $\mathcal{G}_T$ is rooted at v, we define each node's children with respect to v_T. That is, node z is among w's children if z is a neighbor of w and $z \notin \phi(w, v_T)$.

Let $\mathcal{G}_T^{v_i \to v_j}$ denote the subtree of $\mathcal{G}_T$ rooted at node v_j with node v_i as parent of v_j (let $\mathcal{G}_T^{v_1\to v_1} = \mathcal{G}_T$). Each node v_i in $\mathcal{G}_T$ will have some number of child subtrees. Some of these subtrees may be identical (i.e., given a realization $R_{\mathcal{G}_T,v}$ of the labeling random process, they would be isomorphic); let k_v denote the number of distinguishable subtrees of node v. We use $\Delta_1^v, \ldots, \Delta_{k_v}^v$ to denote the *number* of each distinct subtree appearing among the child subtrees of node v (recall children are defined with respect to v_T). For example, node v in graph $\mathcal{G}_T$ in Figure 21 (left panel) has $\Delta_1^v = 1$ and $\Delta_2^v = 2$, since the first of v's child subtrees is equal only to itself, and the second (middle) subtree is isomorphic to the subtree on the right. If there exists a neighboring, unvisited subtree rooted at a *parent* of v, then we say $\Delta_0^v = 1$ (by definition, there will only be one such subtree, and it cannot be equal to any child subtrees because G_T is balanced). Otherwise, we say $\Delta_0^v = 0$. This distinction becomes relevant if $v \neq v_T$. For example the figure below shows a tree that is rooted at $w \neq v_T$. In computing $H(\mathcal{G}_T, w)$, we have $\Delta_0^w = 1$ because there is an unvisited branch from w that contains v_T, and $\Delta_1^w = 2$ because both child subtrees of w are identical.



Let γ^v denote the unvisited neighbors of node v in $\mathcal{G}_T$. We give a recursive expression for computing $H(\mathcal{G}_T, v)$.

Lemma 9.2:

$$H(\mathcal{G}_T, v) = \binom{d_v}{\Delta_0^v, \Delta_1^v, \dots, \Delta_k^v} \prod_{w \in \gamma^v} H(\mathcal{G}_T^{v \to w}, w). \quad (24)$$

Proof: We show this by induction on the depth λ of $\mathcal{G}_T$ (rooted at v). For $\lambda = 1$, $\mathcal{G}_T$ has a node v and d_v neighbors. Every realization of the random breadth-first labeling of $\mathcal{G}_T$ will yield an identical graph since the neighbors of v are indistinguishable, so $H(\mathcal{G}_T, v) = \binom{d_v}{d_v} = 1$.

Now suppose equation (24) holds for all graph-node pairs (G_T, v) with $\lambda < \lambda_o$; we want to show that it holds for $\lambda = \lambda_o$. We can represent $\mathcal{G}_T$ as a root node v and d_v subtrees: $\mathcal{G}_T^{v \to w}$ for $w \in \gamma^v$. Since each subtree has depth at most $\lambda_o - 1$, we can compute $H(\mathcal{G}_T^{v \to w}, w)$ for each subtree $\mathcal{G}_T^{v \to w}$ using equation (24) (from the inductive hypothesis).

Suppose we impose (any) valid labeling on $\mathcal{G}_T$ starting from v; we refer to the labeled graph as $R_{\mathcal{G},v}$. Given $R_{\mathcal{G},v}$, we order the subtrees of a node in ascending order of their numeric labels. For any fixed ordering of the d_v subtrees of v, we have $\prod_{w \in \gamma^v} H(\mathcal{G}_T^{v \to w}, w)$ nonidentical labelings of $\mathcal{G}_T$ that respect the ordering of subtrees and are isomorphic to any given realization $R_{\mathcal{G}_T,v}$. At most, there can be $d_v!$ arrangements of the subtrees. However, some of the subtrees are isomorphic, so this value over-counts the number of distinct arrangements. That is, switching the order of two nonidentical, isomorphic subtrees is the same as preserving the order and changing both subtrees to the appropriate nonidentical, isomorphic subtree; this is already accounted for in the product $\prod_{w \in \gamma^v} H(G_T^{v \to w}, w)$. $\Delta_j^v!$ of the $d_v!$ permutations of v's subtrees permute the jth unique subtree with isomorphisms of itself. As such, the non-redundant number of different arrangements of the subtrees of node v is $\frac{d_v!}{\Delta_0^v!, \Delta_1^v! \dots \Delta_{k_v}^v!} = \binom{d_v}{\Delta_0^v, \Delta_1^v, \dots, \Delta_{k_v}^v}$. This gives the expression in equation (24). ■

Step 2. We want to show that

$$\mathbb{P}(C_{\mathcal{G}_T,v}) = \left(\prod_{v \in G_T} \mathbb{P}_D(d_v) \right) \frac{H(\mathcal{G}_T, v_T) |\chi_{\mathcal{G}_T,v}|}{d_{v_T} \prod\limits_{\substack{w \in \phi(v, v_T) \\ \setminus \{v, v_T\}}} (d_w - 1)}.$$

Since $\mathbb{P}(C_{\mathcal{G}_T,v}) = \left(\prod_{v \in G_T} \mathbb{P}_D(d_v) \right) Q(\mathcal{G}_T, v) H(\mathcal{G}_T, v)$, this is equivalent to showing that

$$\begin{aligned} \frac{H(\mathcal{G}_T, v)}{H(\mathcal{G}_T, v_T)} &= \frac{|\chi_{\mathcal{G}_T,v}|}{Q(\mathcal{G}_T, v) d_{v_T} \prod\limits_{\substack{w \in \phi(v, v_T) \\ \setminus \{v, v_T\}}} (d_w - 1)} \\ &= \frac{d_v}{d_{v_T}} |\chi_{\mathcal{G}_T,v}|. \end{aligned}$$

The expressions for $H(\mathcal{G}_T, v_T)$ and $H(\mathcal{G}_T, v)$ differ in that the former starts at the virtual source and counts all subtrees by "trickling down" the tree (i.e., $\Delta_0^w = 0$ for all $w \in G_T$), whereas the latter progresses from an infected leaf v to the virtual source, then recurses over the remaining, unvisited subtrees of v_T. Let P_i denote the ith node in the path from v to v_T, which has length ℓ. We get

$$\begin{aligned} H(\mathcal{G}_T, v) = \binom{d_{P_1}}{1, d_{P_1} - 1} \times \\ \binom{d_{P_2} - 1}{1, \Delta_1^{P_2} - 1, \dots, \Delta_{k_{P_2}}^{P_2}} \prod_{w \in \gamma^{P_2} \setminus \{P_1, P_3\}} H(\mathcal{G}_T^{P_2 \to w}, w) \times \\ \dots \\ \binom{d_{P_{\ell-1}} - 1}{1, \Delta_1^{P_{\ell-1}} - 1, \dots, \Delta_{k_{P_{\ell-1}}}^{P_{\ell-1}}} \prod_{\substack{w \in \gamma^{P_{\ell-1}} \\ \setminus \{P_{\ell-2}, P_\ell\}}} H(\mathcal{G}_T^{P_{\ell-1} \to w}, w) \times \\ \binom{d_{P_\ell} - 1}{\Delta_1^{P_\ell} - 1, \dots, \Delta_{k_{P_\ell}}^{P_\ell}} \prod_{w \in \gamma^{P_\ell} \setminus \{P_{\ell-1}\}} H(\mathcal{G}_T^{P_\ell \to w}, w). \end{aligned}$$

where each line corresponds to the terms that result from recursively moving up the path from $v = P_1$ to $v_T = P_\ell$. Similarly, we have

$$\begin{aligned} H(\mathcal{G}_T, v_T) = \binom{d_{P_1} - 1}{d_{P_1} - 1} \times \\ \binom{d_{P_2} - 1}{\Delta_1^{P_2}, \dots, \Delta_{k_{P_2}}^{P_2}} \prod_{w \in \gamma^{P_2} \setminus \{P_1, P_3\}} H(\mathcal{G}_T^{P_2 \to w}, w) \times \\ \dots \\ \binom{d_{P_{\ell-1}} - 1}{\Delta_1^{P_{\ell-1}}, \dots, \Delta_{k_{P_{\ell-1}}}^{P_{\ell-1}}} \prod_{\substack{w \in \gamma^{P_{\ell-1}} \\ \setminus \{P_{\ell-2}, P_\ell\}}} H(\mathcal{G}_T^{P_{\ell-1} \to w}, w) \times \\ \binom{d_{P_\ell}}{\Delta_1^{P_\ell}, \dots, \Delta_{k_{P_\ell}}^{P_\ell}} \prod_{w \in \gamma^{P_\ell} \setminus \{P_{\ell-1}\}} H(\mathcal{G}_T^{P_\ell \to w}, w). \end{aligned}$$

Here we have expanded the expression in terms of the path from v to v_T to make simplification clearer, where v is the node over which we previously computed $H(\mathcal{G}_T, v)$. Computing the ratio of $H(\mathcal{G}_T, v)$ to $H(\mathcal{G}_T, v_T)$, all the rightmost products of each line cancel. We are left with the ratio of the combinatorial expressions, which simplify to

$$\begin{aligned} \frac{H(\mathcal{G}_T, v)}{H(\mathcal{G}_T, v_T)} &= \frac{d_{P_1}}{d_{P_\ell}} \Delta_1^{P_2} \dots \Delta_1^{P_{\ell-1}} \Delta_1^{P_\ell} \\ &= \frac{d_v}{d_{v_T}} \Delta_1^{v+1} \dots \Delta_1^{v_T - 1} \Delta_1^{v_T}. \end{aligned}$$

Each Δ_1 denotes the number of child subtrees that are identical to the one containing v, for a given root. As such, the product of Δs above is precisely the number of candidates in the class being considered, or $|\chi_{\mathcal{G}_T,v}|$. That is, since they are indistinguishable in the unlabelled graph, they generate the same family $C_{\mathcal{G}_T,v}$.

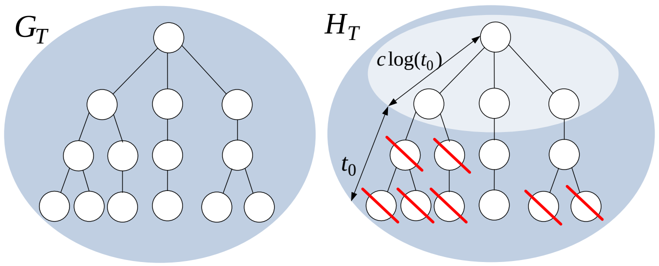


Fig. 23: Pruning of a snapshot. In this example, the distribution D allows nodes to have degree 2 or 3, so we prune all descendants of nodes with degree 3 that are more than $c\log(t_0)$ hops from the root. In this example, $p_1(f_1 - 1) < 1$ and the pruned random process eventually goes extinct.

Step 3. We have

$$\begin{aligned}
\sum_{v\in\mathcal{C}_{\mathcal{G}_T}} \mathbb{P}(C_{\mathcal{G}_T,v}) = & \sum_{v\in\mathcal{C}_{\mathcal{G}_T}} \left(\prod_{w\in G_T} \mathbb{P}_D(d_w)\right) H(\mathcal{G}_T, v_T) \\
& \times \frac{|\chi_{\mathcal{G}_T,v}|}{d_{v_T}\prod_{w\in\phi(v,v_T)\setminus\{v,v_T\}}(d_w-1)} \\
= & \left(\prod_{w\in G_T} \mathbb{P}_D(d_w)\right) H(\mathcal{G}_T, v_T)\times \\
& \sum_{v\in\mathcal{C}_{\mathcal{G}_T}} \frac{|\chi_{\mathcal{G}_T,v}|}{d_{v_T}\prod_{w\in\phi(v,v_T)\setminus\{v,v_T\}}(d_w-1)} \\
= & \left(\prod_{w\in G_T} \mathbb{P}_D(d_w)\right) H(\mathcal{G}_T, v_T)\times \\
& \sum_{v\in\partial G_T} \frac{1}{d_{v_T}\prod_{w\in\phi(v,v_T)\setminus\{v,v_T\}}(d_w-1)}
\end{aligned} \tag{25}$$

where equation (25) follows because every leaf in the graph is a candidate source in exactly one class. We wish to show this last summation sums to 1. Consider a random process over G_T. The process starts at the virtual source v_T, and in each timestep it moves one hop away from v_T. It chooses among the (unvisited) children of a node uniformly at random. At time T, the process is necessarily at one of the leaves of G_T, and the probability of landing at a particular leaf v is precisely $\frac{1}{d_{v_T}\prod_{w\in\phi(v,v_T)\setminus\{v,v_T\}}(d_w-1)}$. Therefore, the sum of this quantity over all leaves $v \in \partial G_T$ is 1.

Step 4. Combining the results from steps 3 and 4, we get that

$$\frac{\mathbb{P}(C_{\mathcal{G}_T,v})/|\chi_{\mathcal{G}_T,v}|}{\sum_{v\in\mathcal{C}_{\mathcal{G}_T}}\mathbb{P}(C_{\mathcal{G}_T,v})} =$$

$$\frac{\left(\prod_{w\in G_T}\mathbb{P}_D(d_w)\right)H(\mathcal{G}_T,v_T)}{\left(\prod_{w\in G_T}\mathbb{P}_D(d_w)\right)H(\mathcal{G}_T,v_T)} \times \frac{\frac{|\chi_{\mathcal{G}_T,v}|/|\chi_{\mathcal{G}_T,v}|}{d_{v_T}\prod_{\substack{w\in\phi(v,v_T)\\ \setminus\{v,v_T\}}}(d_w-1)}}{\frac{1}{d_{v_T}\prod_{w\in\phi(v,v_T)\setminus\{v,v_T\}}(d_w-1)}}.$$

E. Proof of Theorem 4.7

To facilitate the analysis, we consider an alternative random process that generates unlabeled graphs G'_T according to the same distribution as G_T (i.e., the infected, unlabeled subgraph embedded in $U(G_D^{(T)})$ from the proof of Theorem 4.5). For a given degree distribution D and a stopping time T, the new process is defined as a Galton-Watson process in which the set of offsprings at the first time step is drawn from D and the offsprings at subsequent time steps are drawn from $D-1$. At time $t = 0$, a given root node v_T draws its degree d_{v_T} from D, and generates d_{v_T} child nodes. The resulting tree now has depth 1. In each subsequent time step, the process traverses each leaf v of the tree, draws its degree from D, and generates $d_v - 1$ children. The random process continues until the tree has depth $T/2$, since under adaptive diffusion, the infected subgraph at even time T has depth $T/2$. Because the probability of detection in equation (9) does not depend on the degrees of the leaves of G_T, the random process stops at depth $T/2$ rather than $T/2+1$. We call the output of this random process G'_T. The distribution of G'_T is identical to the distribution of the previous random process imposed on G_T, which follows from equation (23) in the proof of Theorem 4.5. We therefore use G_T to denote the resulting output in the remainder of this proof.

Distribution D is a multinomial distribution with support $\boldsymbol{f} = (f_1, \ldots, f_\eta)$ and probabilities $\boldsymbol{p} = (p_1, \ldots, p_\eta)$. Without loss of generality, we assume $2 \le f_1 < \ldots < f_\eta$. Let μ_D denote the mean number of *children* generated by D:

$$\mu_D = \sum_{i=1}^{\eta} p_i(f_i - 1).$$

There are two separate classes of distributions, which we deal with as separate cases.

Case 1: When $p_1(f_1 - 1) > 1$, we claim that with high probability, there exists a leaf node v in ∂G_T such that on the unique path from the root v_T to this leaf v, all nodes in this path have the minimum degree f_1, except for a vanishing fraction. To prove this claim, consider a different graph H_T derived from G_T by pruning large degree nodes:

1) For a fixed, positive c, find t_0 such that $T/2 = t_0 + c\log(t_0)$.
2) Initialize H_T to be identical to G_T.
3) For each node $v \in H_T$, if the hop distance $\delta_H(v, v_T) \le c\log(t_0)$, do not modify that node.
4) For each node $v \in H_T$, if the hop distance $\delta_H(v, v_T) > c\log(t_0)$ and $d_v > f_1$, prune out all the children of v, as well as all their descendants (Figure 23).

We claim that this pruned process survives with high probability. The branching process that generates H_T is equivalent to a Galton-Watson process that uses distribution $D - 1$ for the first $c\log(t_0)$ generations, and a different degree distribution $D' - 1$ for the remaining generations; D' has support $\boldsymbol{f}' = (f_1, 1)$, probability mass $\boldsymbol{p}' = (p_1, 1 - p_1)$, and mean number of children $\mu_{D'} = p_1(f_1 - 1)$.

Note that $f_1 \ge 3$ by the assumption that $p_1(f_1 - 1) > 1$. Hence, the inner branching process up to $c\log t_0$ has probability of extinction equal to 0. This means that at a hop

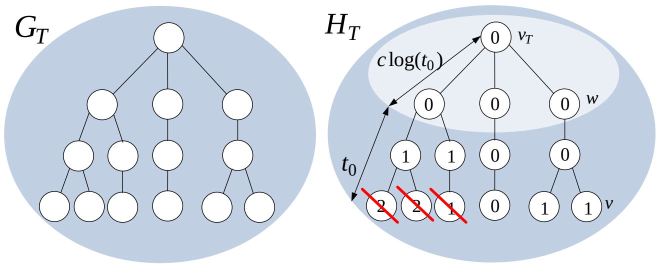


Fig. 24: Pruning of a snapshot using multiple types. In this example, the distribution D allows nodes to have degree 2 or 3. We take $t_0 = 2$ and $r = 0.5$, so all descendants of nodes with type $rt_0 = 1$ are pruned.

distance of t_0 from v_T, there are at least $(f_1-1)^{c\log(t_0)}$ nodes. Each of these nodes can be thought of as the source of an independent Galton-Watson branching process with degree distribution $D'-1$. By the properties of Galton-Watson branching processes ([56], Thm. 6.1), since $\mu_{D'} > 1$ by assumption, each independent branching process' asymptotic probability of extinction is the unique solution of $g_{D'}(s) = s$, for $s \in [0,1)$, where $g_{D'}(s) = p_1\, s^{f_1-1} + (1-p_1)$ denotes the probability generating function of the distribution D'. Call this solution $\theta_{D'}$. The probability of any individual Galton-Watson process going extinct in the first generation is exactly $1-p_1$. It is straightforward to show that $g_{D'}(s)$ is convex, and $g_{D'}(1-p_1) > 1-p_1$, which implies that the probability of extinction is nondecreasing over successive generations and upper bounded by $\theta_{D'}$. Then for the branching process that generates H_T, the overall probability of extinction (for a given time T) is at most $\theta_{D'}^{(f_1-1)^{c\log t_0}}$. Increasing the constant c therefore decreases the probability of extinction. If there exists at least one leaf at depth T (i.e., extinction did not occur), then there exists at least one path in H_T of length $t_0 - c\log t_0$ in which every node (except possibly the final one) has the minimum degree f_1. This gives

$$\frac{\log(\Lambda_{H_T})}{T/2} \le \frac{t_0\log(f_1-1)+c\log(t_0)\log(f_\eta-1)}{t_0+c\log(t_0)} \qquad (26)$$

$$\le \log(f_1-1)+\frac{c\log t_0}{t_0}\log\frac{f_\eta-1}{f_1-1}, \qquad (27)$$

with probability at least $1-\theta_{D'}^{(f_1-1)^{c\log t_0}} = 1-\theta_{D'}^{t_0^{c\log(f_1-1)}} = 1-e^{-C_{D'}t_0}$, where $C_{D'} = \log(\theta_{D'})$ and the upper bound in equation (26) comes from assuming all the interior nodes have maximum degree f_η. Since H_T is a subgraph of a valid snapshot G_T, there exists a path in G_T from the virtual source v_T to a leaf of the tree where the hop distance of the path is exactly $T/2$, and at least t_0 nodes have the minimum degree f_1. Since the second term in equation (27) is $o(t_0)$, the claim follows. The lower bound $\log(\Lambda_{H_T})/(T/2) \ge \log(f_1-1)$ holds by definition. Therefore, for any $\delta > 0$, by setting T (and consequently, t_0) large enough, we can make the second term in equation (27) arbitrarily small. Thus, for $T \ge C'_{D,\delta}$, where $C'_{D,\delta}$ is a constant that depends only on the degree distribution and δ, the result holds.

Case 2: Consider the case when $p_1(f_1-1) \le 1$. By the properties of Galton-Watson branching processes ([56], Thm. 6.1), the previous pruned random process that generated graphs H_T goes extinct with probability approaching 1. This implies that with high probability there is no path from the root to a leaf that consists of only minimum degree nodes.

Instead, we introduce a Galton-Watson process with multiple types, derived from the original process. Our approach is to assign a numeric *type* to each node in G_T according to the number of non-minimum-degree nodes in the unique path between that node and the virtual source. If a node's path to v_T contains too many nodes of high degree, then we prune the node's descendants. The challenge is to choose the smallest pruning threshold that still ensures the pruned tree will survive with high probability. Knowing this threshold allows us to precisely characterize Λ_{G_T} for most of the instances.

To simplify the discussion, we start by considering a special case in which D allows nodes to take only two values of degrees, i.e., $\eta = 2$. We subsequently extend the results for $\eta = 2$ to larger, finite values of η. With a slight abuse of a notation, consider a new random process H_T derived from G_T by pruning large degree nodes in the following way:

1) For a fixed, positive c, find t_0 such that $T/2 = t_0 + c\log(t_0)$.
2) Initialize H_T to be identical to G_T.
3) For each node $v \in H_T$, if the hop distance $\delta_H(v, v_T) \le c\log(t_0)$, do not modify that node, and assign it type 0.
4) For each node $v \in H_T$, if the hop distance $\delta_H(v, v_T) > c\log(t_0)$, assign v a type ξ_v, which is the number of nodes in $\phi(w,v)\setminus\{v\}$ that have the maximum possible degree f_2, where w is the closest node in H_T to v such that $\delta_H(w, v_T) \le c\log(t_0)$ (Figure 24).
5) Given a threshold $r \in (0,1)$, if a node v has type $\xi_v \ge rt_0$, prune out all the descendants of v. For example, in Figure 24, if $t_0 = 2$ and the threshold is $r = 0.5$, we would prune out all descendants of nodes with $\xi_v \ge 1$.

We show that for an appropriately-chosen threshold r, this pruned tree survives with high probability. By choosing the smallest possible r, we ensure that Λ_{H_T} consists (in all but a vanishing fraction of nodes) of a fraction r nodes with maximum degree, and $(1-r)$ of minimum degree. This allows us to derive the bounds on $\log(\Lambda_{H_T})/(T/2)$ stated in the claim, which hold with high probability.

Let $k \equiv rt_0$. The process that generates H_T is equivalent to a different random branching process that generates nodes in the following manner: set the root's type $\xi_{v_T} = 0$. At time $t = 0$, the root v_T draws a number of children according to distribution D, and generates d_{v_T} children, all type 0. Each leaf generates type 0 children according to child degree distribution $D-1$ until $c\log(t_0)$ generations have passed. At that point, each leaf v in this branching process (which necessarily has type 0) reproduces as follows: if its type $\xi_v > k$, then v does not reproduce. Otherwise, it either generates (f_1-1) children with probability p_1, each with state ξ_v, or it generates (f_2-1) children with probability p_2, each with state ξ_v+1. This continues for t_0 generations. Mimicking the notation from Case 1, we use D' to denote the distribution that gives rise to this modified, multi-type random process (in the final t_0 generations); this is a slight abuse of notation since the branching dynamics are multi-type, not defined by realizations of i.i.d. degree random variables.

Lemma 9.3: Consider a Galton-Watson branching process with child degree distribution $D-1$, where each node has at least one child with probability 1, and $\mu_{D-1} > 1$. Then the number of leaves in generation t, $Z^{(t)}$, satisfies the following:

$$Z^{(t)} \geq e^{C_\ell t}$$

with probability at least $1 - e^{C'_\ell t}$, where both C_ℓ and C'_ℓ are constants that depend on the degree distribution.
(Proof in Section IX-E1)

The first $c\log(t_0)$ generations ensure that with high probability, we have at least $e^{C_\ell \log t_0}$ independent multi-type Galton-Watson processes originating from the leaves of the inner subgraph; this follows from Lemma 9.3. Here we have encapsulated the constant c from the first $c\log(t_0)$ generations in the constant C_ℓ. For example, in Figure 24, there are 3 independent Galton-Watson processes starting at the leaves of the inner subgraph. We wish to choose r such that the expected number of new leaves generated by *each* of these processes, at *each* time step, is large enough to ensure that extinction occurs with probability less than one. For brevity, let $\alpha \equiv p_1(f_1-1)$ and let $\beta \equiv p_2(f_2-1)$. Let $x^{(t)}$ denote the $(k+1)$-dimensional vector of the expected number of leaves generated with each type from 0 to k in generation t. This vector evolves according to the following $(k+1)\times(k+1)$ transition matrix M:

$$x^{(t+1)} = x^{(t)} \underbrace{\begin{bmatrix} \alpha & \beta & & \\ & \ddots & & \\ & \ddots & \alpha & \beta \\ & & & 0 \end{bmatrix}}_{M} .$$

The last row of M is 0 because a node with type k does not reproduce. Since the root of each process always has type 0, we have $x^{(0)} = \boldsymbol{e}_1$, where $\boldsymbol{e}_1$ is the indicator vector with a 1 at index 1 and zeros elsewhere.

Let $Z^{(t)}$ denote the expected number of new leaves created in generation t. This gives

$$\mathbb{E}[Z^{(t)}] = \boldsymbol{e}_1 M^t \mathbb{1}^{\intercal}_{(k+1)}, \tag{28}$$

where $^{\intercal}$ denotes a transpose, and $\mathbb{1}_{(k+1)}$ is the $(k+1)$ all-ones vector. When $t < k$, this is a simple binomial expansion of $(\alpha+\beta)^t$. For $t \geq k$, this is a truncated expansion up to k:

$$\mathbb{E}[Z^{(t)}] = \sum_{i=0}^{k} \binom{t}{i} \alpha^{t-i}\beta^i. \tag{29}$$

We seek the necessary and sufficient condition on r for non-extinction, such that $(1/t)\log(\mathbb{E}[Z^{(t)}]) > 0$. Consider a binomial random variable W with parameter $\beta/(\alpha+\beta) = \beta/\mu_D$ and t trials. Equation (29) implies that for large t,

$$\begin{aligned} \mathbb{E}[Z^{(t)}] &= (\alpha+\beta)^t\, \mathbb{P}(W \leq k). \\ &= \mu_D^t \, \exp\Big\{ -t\, D_{\mathrm{KL}}\Big(r \,\|\, \frac{\beta}{\mu_D}\Big) + o(t)\Big\} \,, \end{aligned} \tag{30}$$

by Sanov's theorem [57]. We wish to identify the smallest r for which $(1/t)\log(\mathbb{E}[Z^{(t)}])$ is bounded away from zero. Such an r is a sufficient (and necessary) condition for the multi-type Galton-Watson process to have a probability of extinction less than 1. To achieve this, we define the following set of r such that equation (30) is strictly positive, for some $\epsilon > 0$:

$$\mathcal{R}_{\alpha,\beta}(\epsilon) \;=\; \big\{\, r \mid \log(\mu_D) \geq D_{\mathrm{KL}}(r \| \beta/\mu_D) + \epsilon \,\big\}\,,$$

Suppose we now choose a threshold $r \in \mathcal{R}_{\alpha,\beta}(\epsilon)$. This is the regime where the modified Galton-Watson process with threshold r has a chance for survival. In other words, the probability of extinction $\theta_{D'}$ is strictly less than one. Precisely, $\theta_{D'}$ is the unique solution to $s = g_{D'}(s)$, where $g_{D'}(s)$ denotes the probability generating function of the described multi-type Galton-Watson process. Using the same argument as in Case 1, we can construct a process where the probability of extinction is asymptotically zero. Precisely, we modify the pruning process such that we do not prune any leaves in the first $c\log(t_0)$ generations. This ensures that with high probability, there are at least $e^{C_\ell \log(t_0)}$ independent multi-type Galton-Watson processes evolving concurrently after time $c\log(t_0)$, each with probability of extinction $\theta_{D'}$. Hence with probability at least $1 - e^{-2C_{D'}t_0}$ (for an appropriate choice of a constant $C_{D'}$ that only depends on the degree distribution D' and the choice of r), the overall process does not go extinct.

Our goal is to find the choice of r with minimum product of degrees $\log(\Lambda_{G_T})/(T/2)$ that survives. We define r_1 as follows:

$$r_1 \equiv \operatorname*{arg\,min}_{r\in\mathcal{R}_{\alpha,\beta}(\epsilon)} \quad (1-r)\log(1-f_1) + r\log(1-f_2).$$

Since $\mathcal{R}_{\alpha,\beta}(\epsilon)$ is just an interval and we are minimizing a linear function with a positive slope, the optimal solution is $r_1 = \inf_{r\in\mathcal{R}_{\alpha,\beta}(\epsilon)} r$. This is a choice that survives with high probability and has the minimum product of degrees. Precisely, with probability at least $1 - e^{-C_{D'}T}$, where $C_{D'}$ depends on D' and ϵ, we have that

$$\frac{\log(\Lambda_{G_T})}{T/2} \leq \langle r_1, f\rangle + \frac{c\log(t_0)}{t_0}\log\left(f_2-1\right)$$

where with a slight abuse of notation, we define $\langle r_1, \boldsymbol{f}\rangle \triangleq (1-r_1)\log(f_1-1) + r_1\log(f_2-1)$. It follows that

$$\begin{aligned} &\frac{\log(\Lambda_{G_T})}{T/2} - \langle r^*, \boldsymbol{f}\rangle \leq \\ &(r_1 - r^*)\log\left(\frac{f_2-1}{f_1-1}\right) + \frac{c\log(t_0)}{t_0}\log\left(f_2-1\right) \end{aligned}$$

By setting ϵ small enough and t_0 large enough, we can make this as small as we want. For any given $\delta > 0$, there exists a positive $\epsilon > 0$ such that the first term is bounded by $\delta/2$. Further, recall that $T/2 = c\log(t_0) + t_0$. For any choice of ϵ, there exists a $t_{D',\epsilon}$ such that for all $T \geq t_{D',\epsilon}$ the vanishing term in equation (30) is smaller than ϵ. For any given $\delta > 0$, there exists a positive $t_{D',\delta}$ such that $T \geq t_{D',\delta}$ implies that the second term is upper bounded by $\delta/2$. Putting everything together (and setting ϵ small enough for the target δ), we get that

$$\mathbb{P}\Big(\frac{\log(\Lambda_{G_T})}{T/2} \geq \langle r^*, \boldsymbol{f}\rangle + \delta\Big) \;\leq\; e^{-C_{D',\delta}T}$$

for all $T \geq C'_{D',\delta}$, where $C_{D',\delta}$ and $C'_{D',\delta}$ are positive constants that only depend on the degree distribution D' and the choice of $\delta > 0$.

For the lower bound, we define the following set of r such that equation (30) is strictly negative:

$$\overline{\mathcal{R}}_{\alpha,\beta}(\epsilon) \;=\; \big\{\, r \mid \log(\mu_D) \leq D_{\mathrm{KL}}(r\|\beta/\mu_D) - \epsilon \,\big\}\,.$$

Choosing $r \in \overline{\mathcal{R}}_{\alpha,\beta}(\epsilon)$ causes extinction with probability approaching 1. Explicitly, $\mathbb{P}(Z^{(t)} \neq 0)$ is the probability of non-extinction at time t, and $\mathbb{P}(Z^{(t)} \neq 0) \leq \mathbb{E}[Z^{(t)}]$. By equation (30), we have

$$\mathbb{E}[Z^{(t)}] \;\;\leq\;\; e^{t(\log(\mu_D) - D_{\mathrm{KL}}(r\|\beta/\mu_D) + o(t))}$$

where $\log(\mu_D) - D_{\mathrm{KL}}(r\|\beta/\mu_D) \leq -\epsilon$. The probability of extinction is therefore at least $1 - \mathbb{E}[Z^{(t)}] \geq 1 - e^{-t(\epsilon + o(t))}$. So defining

$$r_2 \equiv \underset{r \in \overline{\mathcal{R}}_{\alpha,\beta}(\epsilon)}{\arg\max} \quad (1-r)\log(1-f_1) + r\log(1-f_2),$$

we have

$$\frac{\log(\Lambda_{G_T})}{T/2} \geq \langle r_2, \boldsymbol{f} \rangle + \frac{c\log(t_0)}{t_0}\log(f_1 - 1)$$

with probability at least $1 - e^{-C_{D',2}T}$ where $C_{D',2}$ is again a constant that depends on D' and ϵ. It again follows that

$$\begin{aligned}&\frac{\log(\Lambda_{G_T})}{T/2} - \langle r^*, \boldsymbol{f} \rangle \geq \\ &(r_2 - r^*)\log\left(\frac{f_2 - 1}{f_1 - 1}\right) + \frac{c\log(t_0)}{t_0}\log\left(f_1 - 1\right)\;\;,\end{aligned}$$

where $r_2 - r^*$ is strictly negative. Again, for any given $\delta > 0$, there exists a positive $\epsilon > 0$ such that the first term is lower bounded by $-\delta/2$, and for any choice of ϵ, there exists a $t_{D',\epsilon}$ such that for all $T \geq t_{D',\epsilon}$ the vanishing term in equation (30) is smaller than ϵ. Note that this ϵ might be different from the one used to show the upper bound. We ultimately choose the smaller of the two ϵ values. For any given $\delta > 0$, there exists a positive $t_{D',\delta}$ such that $T \geq t_{D',\delta}$ implies that the second term is lower bounded by $-\delta/2$. Putting everything together (and setting ϵ small enough for the target δ), we get that

$$\mathbb{P}\Big(\frac{\log(\Lambda_{G_T})}{T/2} \leq \langle r^*, \boldsymbol{f} \rangle - \delta\Big) \;\leq\; e^{-C_{D',\delta}T}$$

for all $T \geq C'_{D',\delta}$, where $C_{D',\delta}$ and $C'_{D',\delta}$ are positive constants that only depend on the degree distribution D' and the choice of $\delta > 0$. This gives the desired result.

We now address the general case for D with support greater than two. We follow the identical structure of the argument. The first major difference is that node types are no longer scalar, but tuples. Each node v's type ξ_v is the $(\eta-1)$-tuple listing how many nodes in the path $\phi(w,v) \setminus \{v\}$ had each non-minimum degree from f_2 to f_η, where w is the closest node to v such that $\delta_H(w, v_T) \leq c\log(t_0)$. Consequently, the threshold $\boldsymbol{r} = [r_1, \ldots, r_{\eta-1}]$ is no longer a scalar, but a vector-valued, pointwise threshold on each element of ξ_v. We let $\boldsymbol{k} = [k_1 = r_1 t_0, \ldots, k_{\eta-1} = r_{\eta-1}t_0]$ denote the time-dependent threshold, and we say $\boldsymbol{k} < \xi_v$ if $k_i < (\xi_v)_i$ for $1 \leq i \leq \eta - 1$. The matrix M is no longer second-order, but a tensor. Equation (28) still holds, except M is replaced with its tensor representation. For brevity, let $\alpha = p_1(f_1 - 1)$ and $\beta_i = p_{i+1}(f_{i+1} - 1)$. Let $\tilde{\beta} = \sum_{i=1}^{\eta-1}\beta_i$. Hence, equation (29) gets modified as

$$\mathbb{E}[Z^{(t)}] = \sum_{i_1=0}^{k_1}\cdots\sum_{i_{\eta-1}=0}^{k_{\eta-1}}\binom{t}{i_1,\ldots,i_{\eta-1}}\alpha^{t-\tilde{\beta}}\beta_1^{i_1}\ldots\beta_{\eta-1}^{i_{\eta-1}}. \tag{31}$$

Now we consider a *multinomial* variable W with parameters β_i/μ_D for $1 \leq i \leq \eta - 1$ and t trials. Note that α/μ_D is the 'failure' probability (corresponding to a node of degree f_1); such events do not contribute to the category count, so the sum of parameters is strictly less than 1. As before, equation (31) can equivalently be written as

$$\begin{aligned}\mathbb{E}[Z^{(t)}] \;&=\; \mu_D^t\,\mathbb{P}(W \leq \boldsymbol{k}) \\ &=\; \mu_D^t\,\exp\Big\{-t\,D_{\mathrm{KL}}\Big(\boldsymbol{r}\,\|\,\Big(\frac{\boldsymbol{\beta}}{\mu_D}\Big)\Big) + o(t)\Big\}\end{aligned} \tag{32}$$

where $\boldsymbol{\beta}/\mu_D$ denotes elementwise division. Once again, we wish to obtain bounds on $\mathbb{P}(W \leq \boldsymbol{k})$. As before, we define the following set of r such that equation (32) is strictly positive, for some $\epsilon > 0$:

$$\mathcal{R}_{\alpha,\boldsymbol{\beta}}(\epsilon) \;=\; \Big\{\,\boldsymbol{r} \mid \log(\mu_D) \geq D_{\mathrm{KL}}(\boldsymbol{r}\|\Big(\frac{\boldsymbol{\beta}}{\mu_D}\Big)) + \epsilon\,\Big\}\,,$$

We now choose a threshold $\boldsymbol{r} \in \mathcal{R}_{\alpha,\beta}(\epsilon)$. Using the same argument as before, we can construct a process where the probability of extinction is asymptotically zero. We again do not prune any leaves in the first $c\log(t_0)$ generations. This ensures that with high probability, there are at least $e^{C_\ell \log(t_0)}$ independent multi-type Galton-Watson processes evolving concurrently after time $c\log(t_0)$, each with probability of extinction $\theta_{D'}$. Hence with probability at least $1 - e^{-2C_{D'}t_0}$ (for an appropriate choice of a constant $C_{D'}$ that only depends on the degree distribution D' and the choice of $\boldsymbol{r}$), the overall process does not go extinct.

We define $\boldsymbol{r}_1$ analogously to the $\eta = 2$ case:

$$\boldsymbol{r}_1 \equiv \underset{\boldsymbol{r} \in \mathcal{R}_{\alpha,\beta}(\epsilon)}{\arg\min} \quad \langle \boldsymbol{r}, \boldsymbol{f} \rangle\;,$$

where we now define $\langle \boldsymbol{r}, \boldsymbol{f} \rangle \equiv (1 - \sum_i r_i)\log(f_1 - 1) + \sum_{j=1}^{\eta-1} r_j \log(f_{j+1} - 1)$. Therefore with probability at least $1 - e^{-C_{D'}T}$, where $C_{D'}$ depends on D' and ϵ, we have that

$$\frac{\log(\Lambda_{G_T})}{T/2} \leq \langle \boldsymbol{r}_1, \boldsymbol{f} \rangle + \frac{c\log(t_0)}{t_0}\log\left(f_\eta - 1\right).$$

It follows that

$$\begin{aligned}&\frac{\log(\Lambda_{G_T})}{T/2} - \langle \boldsymbol{r}^*, \boldsymbol{f} \rangle \leq \\ &\sum_{j=1}^{\eta-1}((r_1)_j - r_j^*)\log\left(\frac{f_{j+1} - 1}{f_1 - 1}\right) + \frac{c\log(t_0)}{t_0}\log\left(f_\eta - 1\right)\;\;.\end{aligned} \tag{33}$$

By setting ϵ small enough and t_0 large enough, we can make this as small as we want. For any given $\delta > 0$, there exists a positive $\epsilon > 0$ such that each term in the summation in equation (33) is bounded by δ/η. Further, recall that $T/2 =$

$c\log(t_0)+t_0$. For any choice of ϵ, there exists a $t_{D',\epsilon}$ such that for all $T \geq t_{D',\epsilon}$ the vanishing term in equation (30) is smaller than ϵ. For any given $\delta > 0$, there exists a positive $t_{D',\delta}$ such that $T \geq t_{D',\delta}$ implies that the second term of equation (33) is upper bounded by δ/η. Putting everything together (and setting ϵ small enough for the target δ), we get that

$$\mathbb{P}\Big(\frac{\log(\Lambda_{G_T})}{T/2} \geq \langle \boldsymbol{r}^*, \boldsymbol{f} \rangle + \delta\Big) \;\leq\; e^{-C_{D',\delta}T}$$

for all $T \geq C'_{D',\delta}$, where $C_{D',\delta}$ and $C'_{D',\delta}$ are positive constants that only depend on the degree distribution D' and the choice of $\delta > 0$.

For the lower bound, we again define a set of r such that equation (30) is strictly negative:

$$\overline{\mathcal{R}}_{\alpha,\boldsymbol{\beta}}(\epsilon) \;=\; \big\{\, \boldsymbol{r} \mid \log(\mu_D) \leq D_{\mathrm{KL}}(\boldsymbol{r} \| \Big(\frac{\boldsymbol{\beta}}{\mu_D}\Big)) - \epsilon \,\big\}\,.$$

Choosing $\boldsymbol{r} \in \overline{\mathcal{R}}_{\alpha,\boldsymbol{\beta}}(\epsilon)$ causes extinction with probability approaching 1. Explicitly, $\mathbb{P}(Z^{(t)} \neq 0)$ is the probability of non-extinction at time t, and $\mathbb{P}(Z^{(t)} \neq 0) \leq \mathbb{E}[Z^{(t)}]$. By equation (30), we have

$$\mathbb{E}[Z^{(t)}] \quad\leq\quad e^{t(\log(\mu_D) - D_{\mathrm{KL}}(\boldsymbol{r}\|\boldsymbol{\beta}/\mu_D) + o(t))}$$

where $\log(\mu_D) - D_{\mathrm{KL}}(\boldsymbol{r}\|\boldsymbol{\beta}/\mu_D) \leq -\epsilon$. The probability of extinction is therefore at least $1 - \mathbb{E}[Z^{(t)}] \geq 1 - e^{-t(\epsilon + o(t))}$. So defining

$$\boldsymbol{r}_2 \equiv \underset{\boldsymbol{r} \in \overline{\mathcal{R}}_{\alpha,\beta}(\epsilon)}{\arg\max} \quad \langle \boldsymbol{r}, \boldsymbol{f} \rangle\;,$$

we have

$$\frac{\log(\Lambda_{G_T})}{T/2} \geq \langle \boldsymbol{r}_2, \boldsymbol{f} \rangle + \frac{c\log(t_0)}{t_0}\log(f_1 - 1)$$

with probability at least $1 - e^{-C_{D',2}T}$ where $C_{D',2}$ is again a constant that depends on D' and ϵ. It follows that

$$\frac{\log(\Lambda_{G_T})}{T/2} - \langle \boldsymbol{r}^*, \boldsymbol{f} \rangle \geq \\ \sum_{j=1}^{\eta-1} ((r_2)_j - r_j^*) \log\left(\frac{f_{j+1}-1}{f_1 - 1}\right) + \frac{c\log(t_0)}{t_0}\log\left(f_\eta - 1\right)\;. \tag{34}$$

where $(r_2)_j - r_j^*$ is strictly negative. Again, for any given $\delta > 0$, there exists a positive $\epsilon > 0$ such that each term in the summation in equation (34) is lower bounded by $-\delta/\eta$, and for any choice of ϵ, there exists a $t_{D',\epsilon}$ such that for all $T \geq t_{D',\epsilon}$ the vanishing term in equation (30) is smaller than ϵ. We again choose the smaller of the two ϵ values from the upper and lower bound. For any given $\delta > 0$, there exists a positive $t_{D',\delta}$ such that $T \geq t_{D',\delta}$ implies that the second term is lower bounded by $-\delta/\eta$. Putting everything together (and setting ϵ small enough for the target δ), we get that

$$\mathbb{P}\Big(\frac{\log(\Lambda_{G_T})}{T/2} \leq \langle \boldsymbol{r}^*, \boldsymbol{f} \rangle - \delta\Big) \;\leq\; e^{-C_{D',\delta}T}$$

for all $T \geq C'_{D',\delta}$, where $C_{D',\delta}$ and $C'_{D',\delta}$ are positive constants that only depend on the degree distribution D' and the choice of $\delta > 0$. This gives the desired result.

1) Proof of Lemma 9.3: If $f_1 > 2$, then the claim follows directly, because each leaf generates at least 2 children in each generation.

If $f_1 = 2$, then for parameters $\rho > 0$ and $\lambda > 0$, we use the Markov inequality to get

$$\begin{aligned} \mathbb{P}(Z^{(t)} \leq \rho) \quad &\leq \quad \mathbb{E}[e^{-\lambda Z^{(t)}}]e^{\rho\lambda} \\ &= \quad g_{D-1}^{(t)}(e^{-\lambda})e^{\rho\lambda}\;, \end{aligned}$$

where $g_{D-1}(s) = \mathbb{E}[e^{s(D-1)}]$ is the probability generating function of $D-1$, and $g_{D-1}^{(t)}(s)$ is the t-fold composition of this function. The goal is to choose parameters ρ and λ such that this quantity approaches zero exponentially fast. The challenge is understanding how $g_{D-1}^{(t)}(e^{-\lambda})$ behaves for a given choice of λ.

Figure 25 illustrates $g_{D-1}(s)$. Because each node always has at least one child, the probability of extinction for this branching process is 0. As such, the probability generating function is convex, with $g_{D-1}(0) = 0$ and $g_{D-1}(1) = 1$. This implies that for any starting point $e^{-\lambda}$, the fixed-point iteration method approaches 0. We characterize the rate at which $g_{D-1}^{(t)}(s_0)$ approaches 0 by separately bounding the rate of convergence in three different regions of s (Figure 25). First, we choose a starting point $s_0 = e^{-\lambda}$. We pick any value $s_1 < 1$, such that the slope is strictly larger than one, i.e. $g'_{D-1}(s_1) > 1$. There may be multiple points that satisfy this property; we can choose any one of them, since it only changes the constant factor in the exponent. Without loss of generality, we assume that $s_0 > s_1$, since otherwise we can start the analysis from the region III. Then region I consists of all $s \in [s_1, s_0]$. To define s_2, we draw a line segment parallel to the diagonal from s_1. The intersection is defined as $(s_2, g_{D-1}(s_2))$. Region II consists of all $s \in [s_2, s_1)$. Finally, we choose a threshold ϵ, below which we say the process has converged. Then region III consists of all $s \in [\epsilon, s_2)$. We wish to identify a time t that guarantees, for a given ϵ and λ, that $g_{D-1}^{(t)}(e^{-\lambda}) \leq \epsilon$.

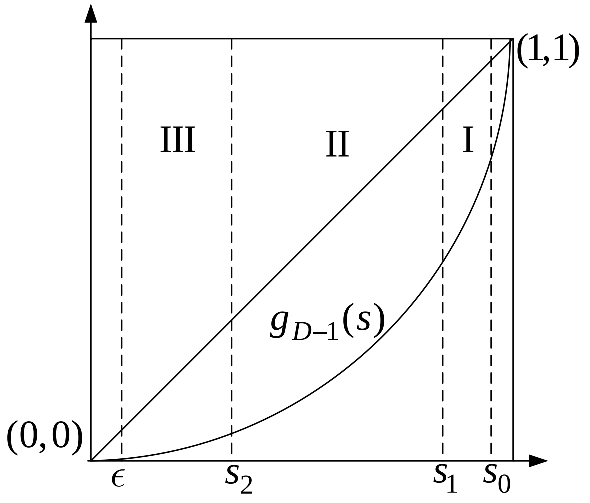


Fig. 25: Regions of the probability generating function, in which we bound the rate of convergence.

To begin, we split the time spent in each region into t_1, t_2, and t_3, with $t_1 + t_2 + t_3 = t$. We first characterize t_1. Note that $g_{D-1}(s_0) \le 1 - g'_{D-1}(s_1)(1-s_0)$ for s_0 in region I. This holds because s_1 has the lowest slope of all points in region I. Applying this recursively, we get that

$$g_{D-1}^{(t_1)}(s) \quad \le \quad \max\left\{1 - g'_{D-1}(s_1)^{t_1}(1-s), g_{D-1}(s_1)\right\}$$

for all s in region I. In region II, we instead upper bound $g_{D-1}(s)$ by the line segment joining $g_{D-1}(s_1)$ and $g_{D-1}(s_2)$. This line has slope 1, giving

$$g_{D-1}^{(t_2)}(s) \le \max\left\{g_{D-1}(s_1) - (s_1 - g_{D-1}(s_1))t_2, g_{D-1}(s_2)\right\}.$$

In region III, we upper bound $g_{D-1}(s)$ by the line $y(s) = g'_{D-1}(s_2)s$. We have that $g_{D-1}(s) < g'_{D-1}(s_2) \cdot s$ for s in region III. Recursing this relation gives

$$g_{D-1}^{(t_3)}(s) \quad \le \quad \max\left\{g'_{D-1}(s_2)^{t_3} \cdot s, \epsilon\right\}.$$

Thus, if $t \ge 3\max\{t_1, t_2, t_3\}$, then $g_{D-1}^{(t)}(e^{-\lambda}) \le \epsilon$. In particular, we choose

$$\begin{aligned} t \quad \ge \quad & 3\max\Big\{\frac{\log((1-g_{D-1}(s_1))/(1-e^{-\lambda}))}{\log(g'_{D-1}(s_1))}, \\ & \frac{g_{D-1}(s_1) - g_{D-1}(s_2)}{s_1 - g_{D-1}(s_1)}, \frac{\log(\epsilon)}{s_2\log(g'_{D-1}(s_2))}\Big\}. \quad (35) \end{aligned}$$

So for sufficiently large t, we have $\mathbb{P}(Z^{(t)} \le \rho) \le \epsilon \cdot e^{\rho\lambda}$. By choosing

$$\epsilon = g'_{D-1}(s_2)^{s_2 t/3},$$

we ensure that the third bound on t is always true, and the other two are constant. Similarly, we choose

$$e^{-\lambda} = 1 - \frac{1-s_2}{g'_{D-1}(s_1)^{t/3}},$$

giving

$$\begin{aligned} \mathbb{P}(Z^{(t)} \le \rho) \quad & \le \quad s_2 \cdot g'_{D-1}(s_2)^{t/3}\left(1 - \underbrace{\frac{1-s_2}{g'_{D-1}(s_1)^{t/3}}}_{B}\right)^{-\rho} \\ & = \quad s_2 \cdot g'_{D-1}(s_2)^{t/3}\left(1-B\right)^{-\frac{1}{B}B\rho} \\ & \le \quad s_2 \cdot g'_{D-1}(s_2)^{t/3} e^{\rho(1-s_2)g'_{D-1}(s_1)^{-t/3}}. \end{aligned}$$

Choosing $\rho = g'_{D-1}(s_1)^{t/3}/(1-s_2)$, we observe that for t larger than the bound in equation (35), the number of leaves is lower bounded by an exponentially growing quantity (ρ) with probability approaching 1 exponentially fast in t.

F. Proof of Proposition 4.9

Number of nodes. T is either even or odd. At each even T, G_T is a ball (defined over a grid graph) centered at the virtual source with radius $T/2$; that is, G_T consists of all nodes whose distance from the virtual source is at most $T/2$ hops. Thus at each successive even T, G_T increases in radius by one. The perimeter of such a ball (over a two-dimensional grid) is $4\frac{T}{2}$. The total number of nodes is therefore $1 + \sum_{i=1}^{T/2} 4i = \frac{1}{2}(T^2 + 2T + 2)$.

When T is odd, there are two cases. Either the virtual source did not move, in which case $N_T = N_{T+1}$ (because all the spreading occurs in one time step), or the virtual source did move, so spreading occurs over two timesteps. In the latter case, the odd timestep adds a number of nodes that is at least half plus one of the previous timestep's perimeter nodes: $N_T \ge N_{T-1} + 2\frac{T-1}{2} + 1 = \frac{1}{2}(T^2 + 2T + 1)$. This is the smaller of the two expressions, so we have $N_T \ge (T+1)^2/2$.

Probability of detection. At each even T, G_T is symmetric about the virtual source. We reiterate that the snapshot adversary can only see which nodes are infected—it has no information about who infected whom.

In order to ensure that each node is equally likely to be the source, we want the distribution of the (strictly positive) distance from the virtual source to the true source to match exactly the distribution of nodes at each viable distance from the virtual source:

$$p^{(t)} = \frac{4}{t(\frac{t}{2}+1)}\begin{bmatrix} 1 \\ 2 \\ \vdots \\ t/2 \end{bmatrix} \in \mathbb{R}^{t/2}. \quad (36)$$

There are $4h$ nodes at distance h from the virtual source, and by symmetry all of them are equally likely to have been the source, giving:

$$\begin{aligned} \mathbb{P}(G_T | v^*, \delta_H(v^*, v_t) = h) \quad & = \quad \frac{1}{4h}p_h^{(t)} \\ & = \quad \frac{1}{t(\frac{t}{2}-1)}, \end{aligned}$$

which is independent of h. Thus all nodes in the graph are equally likely to have been the source. The claim is that by choosing $\alpha(t,h)$ according to equation (16), we satisfy the distribution in equation (36).

Protocol 5 Grid adaptive diffusion

```
Input: grid contact network $G = (V, E)$, source $v^*$, time $T$
Output: set of infected nodes $V_T$
$V_0 \leftarrow \{v^*\}$, $h \leftarrow 0$, $v_0 \leftarrow v^*$
$\mathcal{K} \leftarrow \{N, S, E, W\}$                    ▷ Cardinality directions
let $k_v(u)$ denote $u$'s direction with respect to $v$
$v^*$ selects one of its neighbors $u$ at random
$V_1 \leftarrow V_0 \cup \{u\}$, $v_1 \leftarrow u$
$h^H = \mathbb{1}_{\{k_v(u)=E\}} - \mathbb{1}_{\{k_v(u)=W\}}$
$h^V = \mathbb{1}_{\{k_v(u)=N\}} - \mathbb{1}_{\{k_v(u)=S\}}$
let $N^K(u)$ represent $u$'s neighbors in directions $K \subseteq \mathcal{K}$
$V_2 \leftarrow V_1 \cup N^{\mathcal{K}}(u) \setminus \{v^*\}$, $v_2 \leftarrow v_1$
$t \leftarrow 3$
for $t \leq T$ do
    $v_{t-1}$ selects a random variable $X \sim U(0, 1)$
    if $X \leq \alpha(t-1, |h^V| + |h^H|)$ then
        for all $v \in N(v_{t-1})$ do
            Infection Message($G$,$v_{t-1}$,$v$,$\{k_v(v_{t-1})\}$, $G_t$)
    else
        $K \leftarrow \emptyset$
        if $h^H < 0$ then
            $K \leftarrow K \cup \{E\}$
        else if $h^H > 0$ then
            $K \leftarrow K \cup \{W\}$
        if $h^V < 0$ then
            $K \leftarrow K \cup \{N\}$
        else if $h^V > 0$ then
            $K \leftarrow K \cup \{S\}$
        $v_{t-1}$ randomly selects $u \in N^{\mathcal{K}\setminus K}(v_{t-1})$
        $h^H = h^H + \mathbb{1}_{\{k_v(u)=E\}} - \mathbb{1}_{\{k_v(u)=W\}}$
        $h^V = h^V + \mathbb{1}_{\{k_v(u)=N\}} - \mathbb{1}_{\{k_v(u)=S\}}$
        $v_t \leftarrow u$
        for all $v \in N^{\mathcal{K}\setminus\{k_{v_{t-1}}(v)}(v_t)\}$ do
            Infection Message($G$,$v_t$,$v$,$\{k_{v_t}(v_{t-1}), k_v(v_t)\}$,$V_t$)
            if $t + 1 > T$ then
                break
            Infection Message($G$,$v_t$,$v$,$\{k_{v_t}(v_{t-1}), k_v(v_t)\}$,$V_t$)
    $t \leftarrow t + 2$
procedure INFECTION MESSAGE($G$,$u$,$v$,$K$,$V_t$)
    if $v \in V_t$ then
        for all $w \in N^{\mathcal{K}\setminus K}(v)$ do
            Infection Message($G$,$v$,$w$,$K$,$G_t$)
    else
        $V_t \leftarrow V_{t-2} \cup \{v\}$
```

The state transition can be represented as the usual $((t/2)+1) \times (t/2)$ dimensional column stochastic matrix:

$$p^{(t+2)} = \begin{bmatrix} \alpha(t,1) & & & \\ 1-\alpha(t,1) & \alpha(t,2) & & \\ & 1-\alpha(t,2) & \ddots & \\ & & \ddots & \alpha(t,t/2) \\ & & & 1-\alpha(t,t/2) \end{bmatrix} p^{(t)}$$

This relation holds because we have imposed the condition that the virtual source never moves closer to the true source. We can solve directly for $\alpha(t,1) = t/(t+4)$, and obtain a recursive expression for $\alpha(t,h)$ when $h > 1$:

$$\alpha(t,h) = \frac{t}{t+4} - \frac{h-1}{h}(1 - \alpha(t,h-1)) \, .$$

We show by induction that this expression evaluates to equation (16). For $h = 2$, we have $\alpha(t,2) = \frac{t}{t+4} - \frac{1}{2}\frac{4}{t+4} = \frac{t-2}{t+4}$. Now suppose that equation (16) holds for all $h < h_0$. We then have

$$\begin{aligned} \alpha(t,h_0) = & \quad \frac{t}{t+4} - \frac{h_0-1}{h_0}(1 - \frac{t-2(h_0-1)}{t+4}) \\ = & \quad \frac{t-2(h_0-1)}{t+4} \, , \end{aligned}$$

which is the claim.

By construction the ML estimator for even T is to choose any node except the virtual source uniformly at random. For odd T, there are two options: either the virtual source stayed fixed or it moved. If the former is true, then spreading occurs in one timestep, so the ML estimator once again chooses a node other than the virtual source uniformly at random. If the virtual source moved, then G_T is symmetric about the edge connecting the old virtual source to the new one. Since the adversary only knows that virtual sources cannot be the true source, the ML estimator chooses one of the remaining $N_T - 2$ nodes uniformly at random. This gives a probability of detection of $1/(N_T - 2)$. The claim follows from observing that $N_T \geq \frac{1}{2}(T+1)^2 - 2 = \frac{(T+3)(T-1)}{2}$.

G. Proof of Proposition 7.1

The control packet at spy node s_1 includes the amount of delay at $s_1 = 0$ and all descendants of s_1, which is the set of nodes $\{-1, -2, \ldots\}$. The control packet at spy node s_2 includes the amount of delay at $s_2 = n+1$ and all descendants of s_2, which is the set of nodes $\{n+2, n+3, \ldots\}$. Given this, it is easy to figure out the whole trajectory of the virtual source for time $t \geq T_1$. Since the virtual source follow i.i.d. Bernoulli trials with probability q, one can exactly figure out q from the infinite Bernoulli trials. Also the direction D is trivially revealed.

To lighten the notations, let us suppose that $T_1 \leq T_2$ (or equivalently $T_{s_1} \leq T_{s_2}$). Now using the difference of the observed time stamps $T_{s_2} - T_{s_1}$ and the trajectory of the virtual source between T_{s_1} and T_{s_2}, the adversary can also figure out the time stamp T_1 with respect to the start of the infection. Further, once the adversary figures out T_1 and the location of the virtual source v_{T_1}, the timestamp T_2 does not provide any more information. Hence, the adversary performs ML estimate using T_1, D and q. Let $B(k,n,q) = \binom{n}{k} q^k (1-q)^{n-k}$ denote the pmf of the binomial distribution. Then, the likelihood can be computed for T_1 as

$$\mathbb{P}^{(\text{adaptive})}_{T_1|V^*,Q,D}\big(t_1 \big| v^*, q, \ell\big) = \left\{ \begin{array}{rl} q\,B(v^* - \frac{t_1}{2} - 2, \frac{t_1}{2} - 2, q)\,\mathbb{I}_{(v^* \in [2+\frac{t_1}{2}, t_1])} & \text{, if } t_1 \text{ even ,} \\ B(v^* - \frac{t_1+3}{2}, \frac{t_1-3}{2}, q)\,\mathbb{I}_{(v^* \in [\frac{t_1+3}{2}, t_1])} & \text{, if } t_1 \text{ odd ,} \end{array} \right. \tag{37}$$

$$\mathbb{P}^{(\text{adaptive})}_{T_1|V^*,Q,D}\big(\, t_1 \big|\, v^*, q, r\big) =$$
$$\begin{cases} 0 & \text{, if } t_1 \text{ even ,} \\ (1-q)\, B(\frac{t_1-1}{2} - v^*, \frac{t_1-3}{2}, q)\, \mathbb{I}_{(v^* \in [1, \frac{t_1-1}{2}])} & \text{, if } t_1 \text{ odd .} \end{cases} \tag{38}$$

This follows from the construction of the adaptive diffusion. The protocol follows a binomial distribution with parameter q until $(T_1 - 1)$. At time T_1, one of the following can happen: the virtual source can only be passed (the first equation in (37)), it can only stay (the second equation in (38)), or both cases are possible (the second equation in (37)).

Given T_1, Q and D, which are revealed under the adversarial model we consider, the above formula implies that the posterior distribution of the source also follows a binomial distribution. Hence, the ML estimate is the mode of a binomial distribution with a shift, for example when t_1 is even, ML estimate is the mode of $2 + (t_1/2) + Z$ where $Z \sim$ Binom$((t_1/2) - 2, q)$. The adversary can compute the ML estimate:

$$\hat{v}_{\text{ML}} = \begin{cases} \frac{T_1+2}{2} + \Big\lfloor q\Big(\frac{T_1-2}{2}\Big)\Big\rfloor & \text{if } T_1 \text{ even, } D = \ell \\ \frac{T_1+3}{2} + \Big\lfloor q\Big(\frac{T_1-1}{2}\Big)\Big\rfloor & \text{if } T_1 \text{ odd, } D = \ell \\ 1 + \Big\lfloor (1-q)\Big(\frac{T_1-1}{2}\Big)\Big\rfloor & \text{if } T_1 \text{ odd, } D = r \text{ .} \end{cases} \tag{39}$$

Together with the likelihoods in equations (37) and (38), this gives

$$\mathbb{P}^{(\text{adaptive})}_{T_1,D|V^*,Q}\big(\, t_1, r, \hat{v}_{\text{ML}} = v^* \big| v^*, q\big) =$$
$$\frac{1}{2}(1-q)\, B\Big(\frac{t_1-1}{2} - v^*, \frac{t_1-3}{2}, q\Big)\, \mathbb{I}_{(\hat{v}_{\text{ML}} = v^*)}\, \mathbb{I}_{(t_1 \text{ is odd})}$$
$$\mathbb{P}^{(\text{adaptive})}_{T_1,D|Q}\big(\, t_1, r, V^* = \hat{v}_{\text{ML}} \big| q\big) =$$
$$\begin{aligned} &= \quad \frac{1}{2\,n}(1-q)\, B\Big(\frac{t_1-1}{2} - \hat{v}_{\text{ML}}, \frac{t_1-3}{2}, q\Big)\, \mathbb{I}_{(t_1 \text{ is odd})} \\ &\leq \quad \frac{(1-q)}{2\,n}\Big(\frac{\sqrt{2}\, \mathbb{I}_{(t_1 \text{ is odd and } t_1 > 3)}}{\sqrt{\frac{t_1-3}{2} q(1-q)}} + \mathbb{I}_{(t_1=3)}\Big) \end{aligned}$$

where $\hat{v}_{\text{ML}} = \hat{v}_{\text{ML}}(t_1, q, r)$ is provided in equation (39), and the bound on $B(\cdot)$ follows from Gaussian approximation (which gives an upper bound $1/\sqrt{2\pi k q(1-q)}$) and Berry-Esseen theorem (which gives an approximation guarantee of $2 \times 0.4748/\sqrt{kq(1-q)}$) [58], for $k = (t_1 - 3)/2$. Marginalizing out $T_1 \in \{3, 5, \ldots, 2\lfloor(n-1)/2\rfloor + 1\}$ and applying an upper bound $\sum_{i=1}^{k} 1/\sqrt{i} \leq 2\sqrt{k+1} - 2 \leq 2\sqrt{k-1} + \sqrt{1/(2(k-1))} - 2 \leq \sqrt{4(k-1)}$, we get

$$\mathbb{P}\big(\, D = r, V^* = \hat{v}_{\text{ML}}, T_1 \text{ is odd} \big| Q = q\big) \leq$$
$$\frac{(1-q)\sqrt{2}}{2\,n\,\sqrt{q(1-q)}}\sqrt{8\left\lfloor\frac{n-1}{2}\right\rfloor} + \frac{1-q}{2\,n} \text{ .}$$

Similarly, we can show that

$$\mathbb{P}\big(\, D = \ell, V^* = \hat{v}_{\text{ML}}, T_1 \text{ is odd} \big| Q = q\big) \leq$$
$$\frac{\sqrt{2}}{2\,n\,\sqrt{q(1-q)}}\sqrt{8\left\lfloor\frac{n-1}{2}\right\rfloor} + \frac{1}{n} \text{ ,}$$

$$\mathbb{P}\big(\, V^* = \hat{v}_{\text{ML}}, T_1 \text{ is even} \big| Q = q\big) \leq$$
$$\frac{q\sqrt{2}}{2\,n\,\sqrt{q(1-q)}}\sqrt{8\left\lfloor\frac{n}{2}\right\rfloor} + \frac{1+q}{2\,n} \text{ ,}$$

Summing up,

$$\mathbb{P}(V^* = \hat{v}_{\text{ML}} | Q = q) \quad \leq \quad \sqrt{\frac{8}{n\,q\,(1-q)}} + \frac{2}{n} \text{ .}$$

Recall Q is uniformly drawn from $[0, 1]$. Taking expectation over Q gives

$$\mathbb{P}(V^* = \hat{v}_{\text{ML}}) \quad \leq \quad \pi\sqrt{\frac{8}{n}} + \frac{2}{n} \text{ ,}$$

where we used $\int_0^1 1/\sqrt{x(1-x)}dx = \arcsin(1) - \arcsin(-1) = \pi$.

Acknowledgment

The authors thank Paul Cuff for helpful discussions and for pointing out the Bayesian interpretation of the Pólya's urn process. This work has been supported by NSF CISE awards CCF-1422278, CCF-1553452, and CCF-1409135, SaTC award CNS-1527754, ARO W911NF-14-1-0220, and AFOSR 556016.

References

[1] A. Narayanan and V. Shmatikov, "De-anonymizing social networks," in *Security and Privacy, Symposium on*. IEEE, 2009, pp. 173–187.

[2] B. Greschbach, G. Kreitz, and S. Buchegger, "The devil is in the metadata? New privacy challenges in decentralised online social networks," in *PERCOM*. IEEE, 2012.

[3] D. Cole, "We kill people based on metadata," *New York Review of Books*, 2013.

[4] "Whisper," http://whisper.sh.

[5] "Yik yak," http://www.yikyakapp.com/.

[6] "Secret," https://www.secret.ly.

[7] D. Shah and T. Zaman, "Rumors in a network: Who's the culprit?" *Information Theory, IEEE Transactions on*, vol. 57, no. 8, pp. 5163–5181, Aug 2011.

[8] P. C. Pinto, P. Thiran, and M. Vetterli, "Locating the source of diffusion in large-scale networks," *Physical review letters*, vol. 109, no. 6, p. 068702, 2012.

[9] Z. Wang, W. Dong, W. Zhang, and C. Tan, "Rumor source detection with multiple observations: Fundamental limits and algorithms," in *ACM SIGMETRICS*, 2014.

[10] B. A. Prakash, J. Vreeken, and C. Faloutsos, "Spotting culprits in epidemics: How many and which ones?" in *ICDM*, vol. 12, 2012, pp. 11–20.

[11] V. Fioriti and M. Chinnici, "Predicting the sources of an outbreak with a spectral technique," *arXiv preprint arXiv:1211.2333*, 2012.

[12] W. Luo, W. P. Tay, and M. Leng, "How to identify an infection source with limited observations," *IEEE Journal of Selected Topics in Signal Processing*, vol. 8, no. 4, pp. 586–597, 2014.

[13] G. Danezis and P. Mittal, "Sybilinfer: Detecting sybil nodes using social networks." in *NDSS*. Internet Society, 2009.

[14] G. Fanti, P. Kairouz, S. Oh, K. Ramchandran, and P. Viswanath, "Metadata-conscious anonymous messaging," *IEEE Transactions on Signal and Information Processing over Networks*, vol. 2, no. 4, pp. 582–594, 2016.

[15] D. Shah and T. Zaman, "Finding rumor sources on random graphs," *arXiv preprint arXiv:1110.6230*, 2011.

[16] K. Zhu and L. Ying, "A robust information source estimator with sparse observations," *Computational Social Networks*, vol. 1, no. 1, p. 3, 2014.

[17] R. Dingledine, N. Mathewson, and P. Syverson, "Tor: The second-generation onion router," DTIC Document, Tech. Rep., 2004.

[18] I. Clarke, O. Sandberg, B. Wiley, and T. Hong, "Freenet: A distributed anonymous information storage and retrieval system," in *Designing Privacy Enhancing Technologies*. Springer, 2001.

[19] R. Dingledine, M. Freedman, and D. Molnar, “The free haven project: Distributed anonymous storage service,” in *Designing Privacy Enhancing Technologies*. Springer, 2001.

[20] M. Freedman and R. Morris, “Tarzan: A peer-to-peer anonymizing network layer,” in *Proc. CCS*. ACM, 2002.

[21] D. Chaum, “The dining cryptographers problem: Unconditional sender and recipient untraceability,” *Journal of cryptology*, vol. 1, no. 1, 1988.

[22] H. Corrigan-Gibbs and B. Ford, “Dissent: accountable anonymous group messaging,” in *CCS*. ACM, 2010.

[23] S. Goel, M. Robson, M. Polte, and E. Sirer, “Herbivore: A scalable and efficient protocol for anonymous communication,” Cornell University, Tech. Rep., 2003.

[24] P. Golle and A. Juels, “Dining cryptographers revisited,” in *Advances in Cryptology-Eurocrypt 2004*. Springer, 2004.

[25] L. von Ahn, A. Bortz, and N. Hopper, “K-anonymous message transmission,” in *Proc. CCS*. ACM, 2003.

[26] H. Corrigan-Gibbs, “Riposte: An anonymous messaging system handling millions of users,” 2014.

[27] D. Shah and T. Zaman, “Rumor centrality: a universal source detector,” in *ACM SIGMETRICS Performance Evaluation Review*, vol. 40, no. 1. ACM, 2012, pp. 199–210.

[28] J. Khim and P.-L. Loh, “Confidence sets for the source of a diffusion in regular trees,” *arXiv preprint arXiv:1510.05461*, 2015.

[29] A. Y. Lokhov, M. Mézard, H. Ohta, and L. Zdeborová, “Inferring the origin of an epidemic with a dynamic message-passing algorithm,” *Physical Review E*, vol. 90, no. 1, p. 012801, 2014.

[30] K. Zhu, Z. Chen, and L. Ying, “Locating the contagion source in networks with partial timestamps,” *Data Mining and Knowledge Discovery*, vol. 30, no. 5, pp. 1217–1248, 2016.

[31] C. Milling, C. Caramanis, S. Mannor, and S. Shakkottai, “Network forensics: Random infection vs spreading epidemic,” in *SIGMETRICS*. ACM, 2012.

[32] ——, “Detecting epidemics using highly noisy data.” in *MobiHoc*, 2013, pp. 177–186.

[33] E. A. Meirom, C. Milling, C. Caramanis, S. Mannor, A. Orda, and S. Shakkottai, “Localized epidemic detection in networks with overwhelming noise.” 2014.

[34] C. Milling, C. Caramanis, S. Mannor, and S. Shakkottai, “On identifying the causative network of an epidemic.” in *Allerton Conference*, 2012, pp. 909–914.

[35] B. A. Miller, N. T. Bliss, P. J. Wolfe, and M. S. Beard, “Detection theory for graphs,” *Lincoln Laboratory Journal*, vol. 20, no. 1, pp. 10–30, 2013.

[36] N. Borisov, G. Danezis, and I. Goldberg, “DP5: A private presence service,” Technical Report, CACR, University of Waterloo, Tech. Rep., 2014.

[37] M. Jawad, P. Serrano-Alvarado, and P. Valduriez, “Protecting data privacy in structured P2P networks,” in *Data Management in Grid and Peer-to-Peer Systems*. Springer, 2009.

[38] P. Winter and S. Lindskog, “How the great firewall of china is blocking Tor,” *FOCI*, 2012.

[39] R. Singel, “Point, click ... eavesdrop: How the FBI wiretap net operates,” *Wired*, 2007.

[40] P. Shadbolt, “FireChat in Hong Kong: How an app tapped its way into the protests,” *CNN*, 2014.

[41] K. Bauer, D. McCoy, B. Greenstein, D. Grunwald, and D. Sicker, “Physical layer attacks on unlinkability in wireless LANs,” in *Privacy Enhancing Technologies*. Springer, 2009, pp. 108–127.

[42] B. Danev, H. Luecken, S. Capkun, and K. El Defrawy, “Attacks on physical-layer identification,” in *Proceedings of the third ACM conference on Wireless network security*. ACM, 2010, pp. 89–98.

[43] C. Karlof and D. Wagner, “Secure routing in wireless sensor networks: Attacks and countermeasures,” *Ad hoc networks*, vol. 1, no. 2, pp. 293–315, 2003.

[44] B. Viswanath, A. Mislove, M. Cha, and K. Gummadi, “On the evolution of user interaction in facebook,” in *Proc. of SIGCOMM WOSN*. ACM, August 2009.

[45] G. Fanti, P. Kairouz, S. Oh, and P. Viswanath, “Spy vs. spy: Rumor source obfuscation,” *SIGMETRICS Perform. Eval. Rev.*, 2015.

[46] J. Ugander, B. Karrer, L. Backstrom, and C. Marlow, “The anatomy of the Facebook social graph,” *arXiv preprint arXiv:1111.4503*, 2011.

[47] N. Johnson and S. Kotz, *Urn models and their application*. Wiley New York, 1977.

[48] G. Giakkoupis, R. Guerraoui, A. Jégou, A. Kermarrec, and N. Mittal, “Privacy-conscious information diffusion in social networks,” pp. 480–496, 2015.

[49] S. Spencer and R. Srikant, “On the impossibility of localizing multiple rumor sources in a line graph,” *ACM SIGMETRICS Performance Evaluation Review*, vol. 43, no. 2, pp. 66–68, 2015.

[50] S. Bubeck, L. Devroye, and G. Lugosi, “Finding adam in random growing trees,” *arXiv preprint arXiv:1411.3317*, 2014.

[51] E. Gwynne and S. Bubeck, “Asymptotic behavior of the eden model with positively homogeneous edge weights,” *arXiv preprint arXiv:1508.05140*, 2015.

[52] V. Jog and P. Loh, “Persistence of centrality in random growing trees,” *arXiv preprint arXiv:1511.01975*, 2015.

[53] V. Jog and P.-L. Loh, “Analysis of centrality in sublinear preferential attachment trees via the crump-mode-jagers branching process,” *IEEE Transactions on Network Science and Engineering*, 2016.

[54] W. Luo, W. P. Tay, and M. Leng, “Rumor spreading and source identification: A hide and seek game,” *arXiv preprint arXiv:1504.04796*, 2015.

[55] W. Luo and W. P. Tay, “Identifying multiple infection sources in a network,” in *Signals, Systems and Computers (ASILOMAR), 2012 Conference Record of the Forty Sixth Asilomar Conference on*. IEEE, 2012, pp. 1483–1489.

[56] T. E. Harris, *The theory of branching processes*. Courier Corporation, 2002.

[57] T. M. Cover and J. A. Thomas, *Elements of information theory*. John Wiley & Sons, 2012.

[58] A. C. Berry, “The accuracy of the gaussian approximation to the sum of independent variates,” *Transactions of the american mathematical society*, vol. 49, no. 1, pp. 122–136, 1941.