Metadata-Conscious Anonymous Messaging

Giulia Fanti[†] FANTI@ILLINOIS.EDU
Peter Kairouz[†] KAIROUZ2@ILLINOIS.EDU
Sewoong Oh[†] SWOH@ILLINOIS.EDU
Kannan Ramchandran[‡] KANNANR@EECS.BERKELEY.EDU
Pramod Viswanath[†] PRAMODV@ILLINOIS.EDU

[†] University of Illinois at Urbana-Champaign, Champaign, IL 61801

[‡] University of California, Berkeley, CA 94701

Abstract

Anonymous messaging platforms like Whisper and Yik Yak allow users to spread messages over a network (e.g., a social network) without revealing message authorship to other users. The spread of messages on these platforms can be modeled by a diffusion process over a graph. Recent advances in network analysis have revealed that such diffusion processes are vulnerable to author deanonymization by adversaries with access to metadata, such as timing information. In this work, we ask the fundamental question of how to propagate anonymous messages over a graph to make it difficult for adversaries to infer the source. In particular, we study the performance of a message propagation protocol called adaptive diffusion introduced in (Fanti et al., 2015). We prove that when the adversary has access to metadata at a fraction of corrupted graph nodes, adaptive diffusion achieves asymptotically optimal source-hiding and significantly outperforms standard diffusion. We further demonstrate empirically that adaptive diffusion hides the source effectively on real social networks.

1. Introduction

Popular social media platforms (Facebook, Twitter, Whatsapp, Kakao) allow users to seamlessly share potentially sensitive content with their friends. The privacy implications of such platforms are gaining attention, leading to the emergence of *anonymous social networks* like Whisper (whi), Yik Yak (yik), Blind (bli) and the now-defunct Secret (sec). These anonymous messaging apps are microblogging services that *hide* message authorship from other users. When a user posts a message, the message passes (without authorship information) to the users' contacts, or *friends*, in an underlying social network. If a message recipient approves a message by pressing the 'like' button, the message is further propagated to the recipient's friends, and so on. The message thus spreads anonymously through the network, in the sense that no single user can learn who authored a message.

A natural question is whether a group of colluding nodes can identify which node in the social network authored a given message. This poses an interesting estimation problem on a random process over a network. Statistical inference plays a crucial role in understanding the propagation of influence and diffusion as studied in the data mining community (Gomez Rodriguez et al., 2010; Bakshy et al., 2011). In particular, recent advances in network analysis such as (Shah & Zaman, 2011; Pinto et al., 2012; Zhu et al., 2015) suggest that a moderately-powerful adversary can infer which node started the message, using limited metadata. However, the designers of anonymous messaging platforms have the freedom to influence content propagation by adding appropriately-chosen delays on top of natural human delays. Our goal is to design messaging protocols that make it provably difficult to infer the initial source node, even for an adversary with infinite computational power.

Adversarial Models. We consider an adversary that has access to the underlying contact network $G(V, E)$. The adversary lacks the resources to monitor all network traffic, but it can collect partial metadata in a number of ways:

One way is to explicitly corrupt some fraction of nodes by bribery or coercion; these corrupted *spy nodes* continuously monitor metadata like message timestamps and relay

Proceedings of the 33^{rd} International Conference on Machine Learning, New York, NY, USA, 2016. JMLR: W&CP volume 48.

IDs; we call this a *spy-based* adversary. This models an adversary using fake or corrupted social media accounts to monitor users (Goldman, 2014).

Alternatively, an adversary could use side channels to collect information on whether each graph node is *infected*, i.e., whether it received the message, at a fixed time; we call this a *snapshot* adversarial model. If an adversary uses spies and a snapshot, we call it a *spy+snapshot* adversary. The snapshot adversary has been well-studied in the literature, for both source identification (Shah & Zaman, 2011) and source obfuscation (Fanti et al., 2015). However, spy-based adversaries have only been studied for source identification (Pinto et al., 2012). In this paper, we focus on spy-based adversaries, and briefly discuss the implications of the spy+snapshot adversary in Sec. 4.

Precisely, under the spy-based model, each node other than the source is a spy with probability p, independently. At some point in time, the source node v^* starts propagating its message over the graph according to a spreading protocol chosen by the platform. Each spy node $s_i \in V$ observes: (a) the time T_{s_i} (relative to an absolute reference) at which it receives the message; (b) the parent node p_{s_i} that relayed the message; and (c) 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, $\hat{v}$. We are interested in both sides of the problem: the adversary and the system designer. For the adversary, we want to design the maximum likelihood estimator of the source for a given messaging protocol. For the system designer, we want to design a spreading mechanism that minimizes the probability of detection, $\mathbb{P}(\hat{v} = v^*)$, for an adversary using the maximum likelihood estimator. This is the focus of this paper.

Spreading mechanisms. A common construction for modeling epidemic propagation over networks is *diffusion*: each node spreads the message to its neighbors according to independent, random delays. 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 of most social media platforms.

Finding a computationally-efficient algorithm for (near-) optimal maximum likelihood (ML) message source inference is an open problem under the spy-based adversarial model, as is the corresponding detection probability analysis. Recent work (Pinto et al., 2012; Zhu et al., 2015) has focused on identifying the message source through heuristic, low-cost algorithms. These findings suggest that a spy-based adversary with metadata can locate the source with high probability under diffusion spreading. Indeed, when the underlying graph is a d-regular tree, we empirically observe that the probability of detection under diffusion increases with time and the degree of the underlying graph (Figure 1). This has poor implications for anonymity; contact networks may have high degree nodes, and the adversary is not time-constrained.

In the diffusion model used to generate Figure 1, each node propagates the message to each of its neighbors independently with probability $q = 0.7$ in each time step. We used the Gaussian estimator from (Pinto et al., 2012), which is suboptimal for this spreading model; as such, the plotted curves are lower bounds on the probability of detection using an ML estimator.

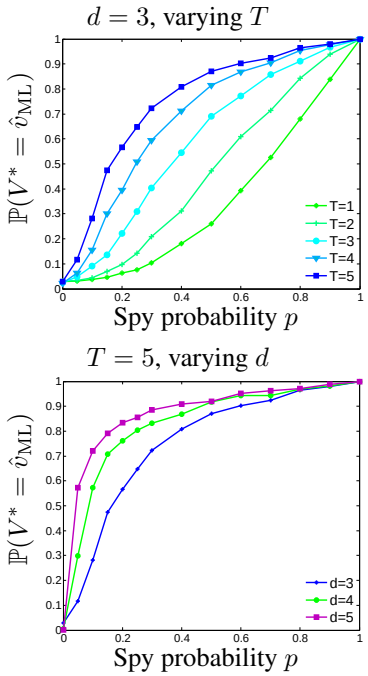

Figure 1. Probability of detection (suboptimal) when a message is spread using diffusion over a d-regular tree. Detection becomes more accurate as time and underlying graph degree increase.

We therefore seek a different spreading model with strong anonymity guarantees when the underlying graph has high degree, and estimation occurs at $T = \infty$. Concretely, the designer is free to use any messaging protocol, under the following scenario. We consider a discrete time system, and at any given time the protocol is allowed to infect any of the adjacent neighbors of an infected node, who is not already infected. The infection can grow at most by 1-hop each time, and we may choose to add additional delays. Under this setting, the analog of standard diffusion is choosing to spread with some fixed probability, independently for each neighbor. *Adaptive diffusion* was introduced in (Fanti et al., 2015) to achieve optimal source

obfuscation under the snapshot model. In this paper, we analyze the anonymity properties of *adaptive diffusion* under the spy-based model. There is no reason to believe *a priori* that adaptive diffusion should perform well against a spy-based adversary with its access to timing information; surprisingly, it does.

Contributions. Our contributions are as follows:

(1) We identify adaptive diffusion as an algorithm that provides strong anonymity guarantees against a *spy-based adversary*. Since (Fanti et al., 2015) contains multiple variants of adaptive diffusion, we identify the specific parameter setting under which it is both analytically tractable and provides strong anonymity guarantees.

(2) Under the spy-based adversarial model and adaptive diffusion spreading, we identify a computationally-efficient algorithm for maximum likelihood source detection when the underlying contact network is infinite and tree-structured (Algorithm 2).

(3) We give a precise analysis of the anonymity properties of adaptive diffusion. Such analysis is currently open for regular diffusion; we provide exact expressions for adaptive diffusion over regular trees (Theorem 1) and a lower bound for regular diffusion (Proposition 3.2), and show that our results are numerically stable for finite, irregular, cyclic social network graphs.

(4) We show that over regular trees, adaptive diffusion has asymptotically optimal hiding guarantees (Proposition 3.1) as the degree of the underlying tree increases. This differs from regular diffusion, whose anonymity properties *degrade* as degree increases. Intuitively, spies near the source provide more information than distant ones; by spreading symmetrically, diffusion ensures that all nearby spies receive the message. Adaptive diffusion instead spreads asymmetrically, thereby preventing most nearby spies from seeing the message early enough to deanonymize.

Related Work. A snapshot-based adversary observes which nodes are infected at a certain time T. When the infection spreads as per *standard diffusion* on a d-regular tree, efficient ML estimators exist for finding the source from the snapshot (Shah & Zaman, 2011). Further, the adversary can identify the source with probability converging to a constant lower-bounded by $1/3$, as the time-to-attack grows. Subsequent work suggests that even under various diffusion models and estimators, source detection with a snapshot is reliable (Wang et al., 2014; Prakash et al., 2012; Fioriti & Chinnici, 2012; Luo et al., 2014; Zhu & Ying, 2014; Milling et al., 2012a; 2013; 2012b; Khim & Loh, 2015).

If we know when the adversary will attack (time T), one solution for hiding the source on a d-regular tree is the following: for the first half of timesteps (until $T/2$), the infection propagates on a line in a randomly chosen direction; for the remaining half, the infection spreads as per diffusion from the end of the line. At time T all nodes in the boundary of the snapshot are equally likely to be the source, by symmetry. However, this *line-and-diffusion* protocol fails to protect the source if the adversary attacks sufficiently before or after time T. *Adaptive diffusion* was proposed to provide strong protection against a snapshot-based adversary (Fanti et al., 2015). At any time T, adaptive diffusion ensures that all nodes are equally likely to have been the source. This provides perfect obfuscation; no adversary can find the source with probability larger than $1/N_T$ where N_T is the number of infected nodes.

When the adversary collects timestamps (and other metadata) from spy nodes, standard diffusion reveals the location of the source (Pinto et al., 2012; Zhu & Ying, 2014; Farajtabar et al., 2015). However, ML estimation is known to be NP-hard (Zhu et al., 2015), and analyzing the probability of detection is also challenging. Figure 1 shows that even with sub-optimal estimators, the source can be effectively identified. Since both snapshot and spy-based adversaries are plausible, we want to go beyond diffusion and line-and-diffusion. A natural question of interest is how to spread a message in order to provide strong protection against both types of adversaries: snapshot and spy-based. 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.

2. Adaptive diffusion

Under standard diffusion, inferring the source is easy (for a snapshot adversary) because the source is typically near the center of the snapshot. In (Fanti et al., 2015), the authors present two protocols that make such inference provably difficult on trees with degree $d > 2$: the 'tree protocol' and a generalization called 'adaptive diffusion'. Intuitively, these protocols randomize the location of the source within each snapshot, making the source difficult to locate.

If the underlying contact network is a tree, then the tree protocol is equivalent to adaptive diffusion for a specific choice of parameters; this parameter choice implies that the source node is always a *leaf* of the infected subgraph. General adaptive diffusion does not make this restriction. Over regular trees and a snapshot adversarial model, adaptive diffusion achieves the minimax optimal probability of detection $1/N_T$, when N_T nodes have received the message at the time of attack (Fanti et al., 2015). Over random irregular trees with non-trivial i.i.d. degree distributions,

the tree protocol was later shown to exhibit a multiplicative gap to optimality, i.e. $\mathbb{P}(\hat{v} = v^*)/(1/N_T)$, that grows exponentially in T (Fanti et al., 2016).

In this work, we consider a spy-based adversarial model and focus on the tree protocol, exploiting its simplicity and asymmetric spreading. Specifically, we show that the tree protocol achieves provably (asymptotically) optimal source obfuscation, significantly improving upon standard diffusion. Moving forward, we use the terms 'tree protocol' and 'adaptive diffusion' interchangeably.

Algorithm 1 Tree protocol (Fanti et al., 2015)

Input: 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 $u_w \leftarrow\uparrow$
    end if
    for all uninfected neighboring nodes z of v **do**
      $V_t \leftarrow V_{t-1} \cup \{z\}$
      $u_z \leftarrow\downarrow$ and $m_z \leftarrow m_v - 1$
    end for
  end for
  $t \leftarrow t + 1$
end for

The tree protocol from (Fanti et al., 2015) is outlined in Algorithm 1; the goal is to build an infected subtree with the true source at one of the leaves at all times. This goal is accomplished by introducing state variables. 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*, $m_w \in \mathbb{N}$, in the infected subtree. 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 $t \to \infty$. The parent metadata did not appear in the original tree protocol (Fanti et al., 2015), and is included purely to facilitate the adversary's source estimation. Even with this extra metadata revealed, the tree protocol achieves asymptotically optimal hiding.

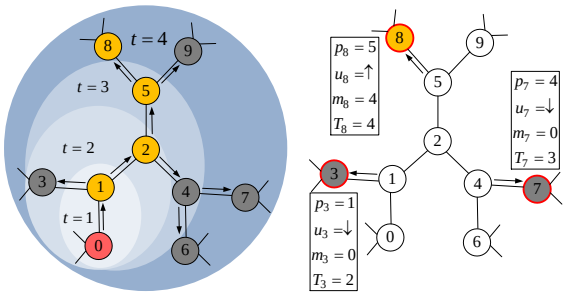

Figure 2. Message spread using the tree protocol from (Fanti et al., 2015) (left), and the information observed by the spy nodes 3, 7, and 8 (right). Timestamps in this figure are absolute, but they need not be.

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 2 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 forwards the message to another neighbor z chosen uniformly at random 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: $(p_{z'} = w,\ u_{z'} = \downarrow,\ m_{z'} = m_w - 1)$. In Figure 2, 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. This protocol ensures that the infected subgraph is a symmetric tree with the true source at one of the leaves (see Algorithm 1).

3. Main results on d-regular trees

We show that adaptive diffusion hides the source better than diffusion over d-regular trees, $d > 2$, and its probability of detection is asymptotically optimal in the degree of the underlying tree. We first present a lower bound on the probability of detection for any choice of spreading protocol.

Proposition 3.1. *When each node in the network is independently chosen to be a spy with probability p, no spreading protocol that infects at least one node can have a probability of detection less than p, i.e.*

$$\min_{\text{protocol}} \max_{\hat{v}} \mathbb{P}(\hat{v} = v^*) \ \geq\ p\,,$$

where the minimization is over all spreading protocols that infect at least one node and the maximization is over all

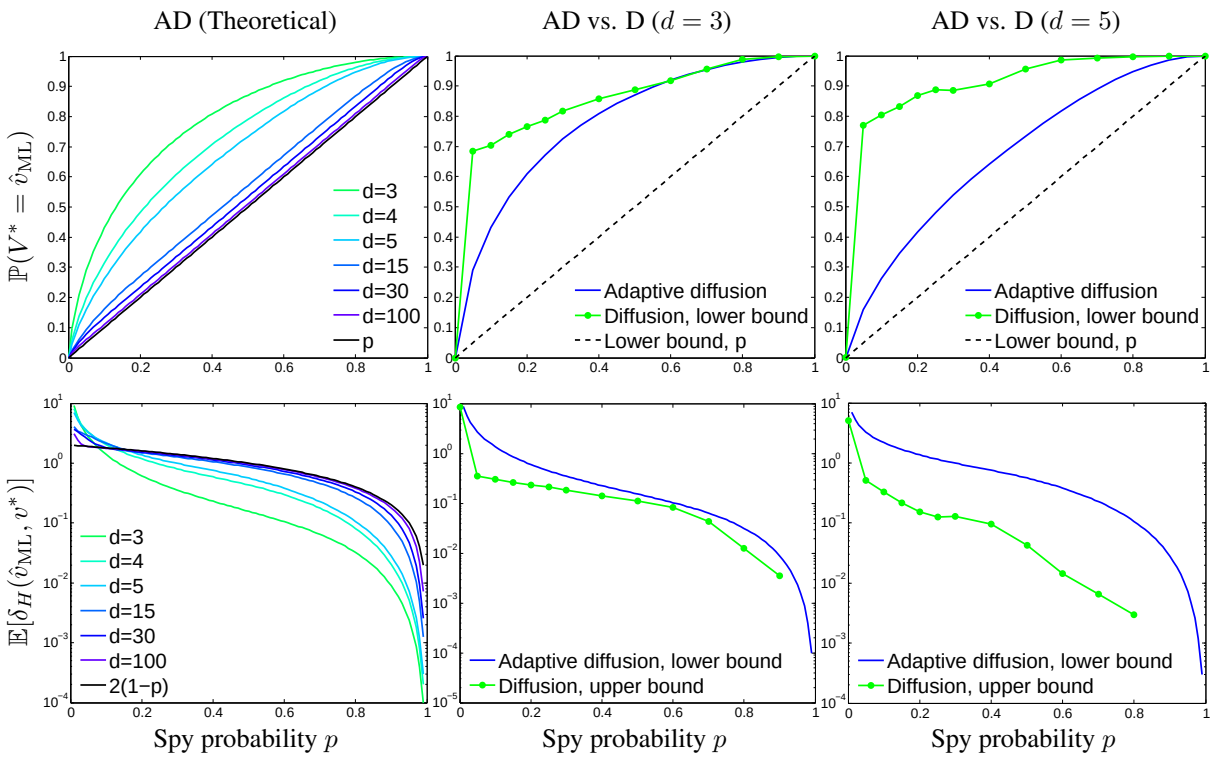

Figure 3. Adaptive diffusion (AD) theoretical performance for varying d (left). Adaptive diffusion improves over standard diffusion (D) and the gap increases as the degree of the underlying contact network increases (center, right).

estimators that are measurable functions over the observed meta-data and the network.

The proof considers a first-spy estimator, which returns as the estimated source the parent (the sender of the message) of the first spy to observe the message. Regardless of spreading mechanism, this estimator returns the true source with probability at least p; with probability p, the first node (other than the true source) to receive the message is a spy. This is illustrated in the top panels of Figure 3 as a fundamental limit. Note that this lower bound is independent of the tree degree, and we expect the bound to be tighter for larger-degree trees. The reason is that if d is larger, then it is more likely that one of the neighbors of the source is a spy.

Standard diffusion. The ML estimator under standard diffusion is computationally intractable, and characterizing the probability of detection achieved by such an estimator is also an open problem. We consider a discrete-time diffusion process, in which each infected node passes the message to each neighbor with probability q in each timestep. As q increases, the variance of the associated geometric delay decreases, revealing the true source with higher probability. To *lower* bound the probability of detection achieved by the best estimator, we consider two heuristic estimators in the numerical experiments: (1) the Gaussian estimator from (Pinto et al., 2012), and (2) the first-spy estimator, which simply returns the parent of the first spy to observe the message. The estimator in (Pinto et al., 2012) is ML when delays are i.i.d. Gaussian, whereas our delays are geometric. We nonetheless expect it to perform well for small p; since the distance between spies will be large, the delay distribution can be approximated by a Gaussian.

Figure 3 compares the probability of detection and expected hop distance for diffusion ($q = 0.7$) using heuristic estimators, against adaptive diffusion using the ML estimator. The lower bound on probability of detection for standard diffusion (top) is the maximum of the simulated Pinto *et al.* estimator (Pinto et al., 2012) and the first-spy estimator; the opposite holds for expected hop distance (bottom). For all p, adaptive diffusion performs better than diffusion, and the gap increases with degree. This effect is sensitive to q for small d, but we show in Section 4 that over real social graphs, the sensitivity to q becomes negligible. We make this precise in the following lower bound:

Proposition 3.2. *Suppose the contact network is a regular tree with degree d. Consider a spy-based adversary and diffusion spreading—that is, in each timestep, each infected node infects each uninfected neighbor independently with probability q. The optimal source estimator achieves*

a detection probability at least

$$\max_{\hat{v}} \mathbb{P}(\hat{v} = v^*) \geq 1 - (1 - qp)^d \ ,$$

where the maximization is over all measurable functions over the observed meta-data and the network.

This bound implies that as degree increases, the probability of detecting the true source of diffusion approaches 1. This proposition can also be proved by considering the first-spy estimator used in Proposition 3.1. We consider all neighbors of v^* that (a) are spies and (b) receive the message at $t = 1$. If there is at least one such node, then the source is identified with probability 1. Each neighbor of v^* meets these criteria with probability pq.

Adaptive diffusion. Unlike standard diffusion, the ML estimator is tractable under adaptive diffusion. Further, we can characterize the probability of detection achieved by this ML estimator precisely, and prove it significantly improves over the standard diffusion and achieves the asymptotically optimal performance.

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 2 (right) illustrates the information observed by each spy node, where spies are outlined in red.

ML estimator under adaptive diffusion. The precise ML estimation algorithm is detailed in Algorithm 2. Because adaptive diffusion has deterministic timing, spies only help the estimator discard candidate nodes. We assume the message spreads for an 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 2, 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.

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 2). For instance, in Figure 2 (right), 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.

Algorithm 2 ML Source Estimator for Algorithm 1

Input: contact network $G = (V, E)$, spy nodes $S = \{s_0, s_1 \ldots\}$ 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 $(p_{s_0}, m_{s_0}, u_{s_0})$.
$\tilde{V} \leftarrow \{v \in V : \ \delta_H(v, s_0) \leq m_{s_0}$ 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}
end for
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**
    $U \leftarrow U \cup \{v\}$
  end if
end for
return $\hat{v}_{\mathrm{ML}}$, drawn uniformly from U

Anonymity properties of adaptive diffusion. Using the described ML estimation procedure, we can exactly compute the probability of detection when adaptive diffusion is run on a d-regular tree.

Theorem 1. *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 i.i.d. with probability p as described in the spy model. Against colluding spies trying 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

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

where $|T_{d,k}| = \frac{(d-1)^k-1}{d-2}$, *and*

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

The proof is included in the Supplemental Materials. Briefly, it computes the probability of detection by conditioning on the lowest-level pivot node, ℓ_{min}. Given a pivot, the probability of detection depends on the number of subtrees rooted at the neighbors of ℓ_{min} containing no spies. Figure 3 illustrates the theoretical probability of detection and lower bound on expected distance from the true source as a function of the spy probability. We make two key observations:

Asymptotically optimal probability of detection: As tree degree d increases, the probability of detection converges to the degree-independent fundamental limit in Proposition 3.1, i.e., $\mathbb{P}(V^* = \hat{v}_{\mathrm{ML}}) = p$. This is in contrast to diffusion, whose probability of detection tends to 1 as $d \to \infty$. The median Facebook user has 200 friends (Smith, 2014), so these asymptotics have practical implications, as we will see in Section 4.

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 $d \to \infty$, $\limsup \mathbb{E}[\delta_H(\hat{v}_{\mathrm{ML}}, v^*)] = 2(1-p)$. This holds because with probability $(1-p)$, the first node is not a spy, but with probability approaching 1 for d large enough, the first node on the spine will be a pivot node. Since the source is always a leaf, the distance from the estimate to the source will be at most 2 with probability approaching $(1-p)$. Figure 3 includes the line $2(1-p)$ for reference, and we observe that as $d \to \infty$, $\mathbb{E}[\delta_H(\hat{v}_{\mathrm{ML}}, v^*)]$ appears to converge precisely to this line. However, for a fixed d, Theorem 1 implies that as $p \to 0$, $\mathbb{E}[\delta_H(\hat{v}_{\mathrm{ML}}, v^*)] \to \infty$.

4. Generalizations

Graphs. Here, we consider irregular, finite graphs that arise in real contact networks. Regardless of spreading protocol, the message always propagates over a tree superimposed on the underlying contact network. The probability of detection over *irregular* trees is therefore tied to performance over general graphs. ML estimation over irregular trees is more straightforward than in (Fanti et al., 2015), because the tree protocol always places the source at a leaf.

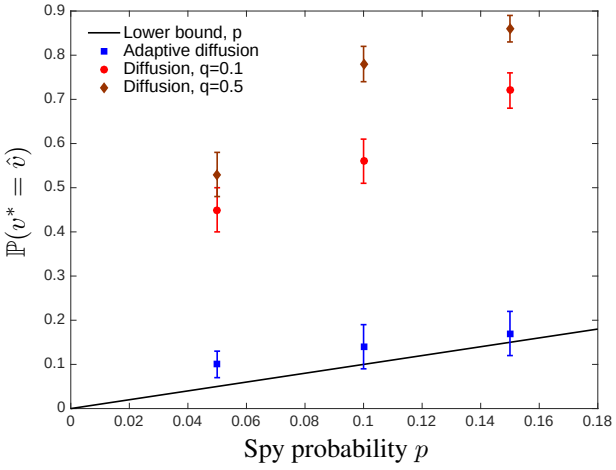

Figure 4. Probability of detection over the Facebook dataset (Viswanath et al., 2009), with standard error.

Proposition 4.1. *Suppose the underlying contact network $G(V, E)$ is an irregular tree with the degree of each node $d_v > 1$. A node $v^* \in V$ starts spreading a message at time $T = 0$ according to Protocol 1. Each node $v \in V$, $v \neq v^*$ is a spy with probability p. Let U denote the set of feasible candidate sources obtained by estimation Algorithm 2. Then the maximum likelihood estimate of v^* given U is* $\hat{v}_{\mathrm{ML}} = \arg\max_{u \in U} \frac{1}{\deg(u)} \prod_{v \in \mathcal{P}(u,\ell_{min}) \setminus \{u,\ell_{min}\}} \frac{1}{\deg(v)-1}$, *where ℓ_{min} is the lowest-level pivot node, $\mathcal{P}(u, \ell_{min})$ is the unique shortest path between u and ℓ_{min}, and $\deg(u)$ denotes the degree of node u. (Proof in Section C, Supplemental Materials.)*

This ML estimator allows us to evaluate adaptive diffusion over real data (social graph connections for 10,000 Facebook users (Viswanath et al., 2009)) against a spy-based adversary. We simulate adaptive and regular diffusion for $q \in \{0.1, 0.5\}$. We evaluate diffusion with the first-spy estimator, and adaptive diffusion with a modification of the ML estimator in Proposition 4.1 that accounts for graph cycles. Figure 4 lists the probability of detection averaged over 200 trials, for $p \leq 0.15$ (at its height, the Stasi employed 11 percent of the population as spies (Koehler, 1999)). Adaptive diffusion hides the source better than diffusion, and its probability of detection is close to the fun-

damental lower bound of p. This is likely because the mean node degree in the dataset is 25, so high-degree asymptotics are significant. While adaptive diffusion does not reach all nodes on a tree, cycles in the Facebook graph allow it to reach 81% of nodes within 20 timesteps.

Adversaries. The spy-based and snapshot adversarial models capture different behavior. The spy-snapshot model considers a combination of both: at a certain time T, the adversary collects both types of metadata and 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 analysis of the probability of detection at T is given in Equation (15) in the Supplementary material, and Figure 6 illustrates the tradeoff between snapshots and spy nodes.

5. Discussion

In this paper, we make the first attempt to bridge the gap between state-of-the-art protocols in (Fanti et al., 2015) for source-hiding under a canonical, simplified, snapshot-based adversarial model, and practical adversarial scenarios. We ask how to spread a message over a graph while preventing a group of colluding nodes from inferring the message source. We show that standard diffusion, modeling existing messaging protocols, exhibits poor hiding properties; i.e. the probability of detection is at least $1-(1-0.5p)^d$, when each node is a spy with probability p on a d-regular tree. However, no protocol can achieve a probability of detection less than p on any connected graph. To bridge this gap, we propose adaptive diffusion—a spreading protocol designed to defend against snapshot adversaries (Fanti et al., 2015)—and show it exhibits asymptotically optimal source-hiding against spy-based adversaries when the graph is a regular tree, and outperforms standard diffusion. We give an exact probability of detection in Theorem 1, illustrated in Figure 3.

A number of open questions remain, including an analysis of adaptive diffusion on more general graph structures. Another important question stems from the fact that in practice, users propagate (and receive) content asynchronously. Analyzing the effect of these delays on anonymity is an interesting open question. Finally, it is not clear that adaptive diffusion is the best scheme to defend against spy-based adversaries, since it was designed to defend against snapshot adversaries. It may be possible to hide the source better by taking a first-principles approach to designing a new spreading mechanism tailored for spy-based adversaries. Beyond rumor spreading, there have been recent advances in finding the first node to start a growing random network (Bubeck et al., 2014; Jog & Loh, 2016; 2015). An interesting direction is to ask how one could grow a random network while preserving the anonymity of the first node under certain constraints or utility.

ACKNOWLEDGMENTS

The authors thank the anonymous reviewers for their helpful comments and Paul Cuff for helpful discussions and for pointing out the Bayesian interpretation of the Polya's urn process. This work has been supported by NSF CISE awards CCF-1422278 and CCF-1553452, and SaTC award CNS-1527754.

References

Team blind. `http://us.teamblind.com/`.

Secret. `https://www.secret.ly`.

Whisper. `http://whisper.sh`.

Yik yak. `http://www.yikyakapp.com/`.

Bakshy, E., Hofman, J. M., Mason, W. A., and Watts, D. J. Everyone's an influencer: quantifying influence on twitter. In *WSDM*, pp. 65–74. ACM, 2011.

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

Fanti, G., Kairouz, P., Oh, S., and Viswanath, P. Spy vs. spy: Rumor source obfuscation. In *ACM SIGMETRICS Perform. Eval. Rev.*, pp. 271–284, 2015.

Fanti, G., Kairouz, P., Oh, S., Ramchandran, K., and Viswanath, P. Rumor source obfuscation on irregular trees. In *ACM SIGMETRICS Perform. Eval. Rev.*, 2016.

Farajtabar, M., Gomez-Rodriguez, M., Du, N., Zamani, M., Zha, H., and Song, L. Back to the past: Source identification in diffusion networks from partially observed cascades. *arXiv preprint arXiv:1501.06582*, 2015.

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

Goldman, David. Cops can use fake facebook accounts to lure suspects, says judge. *CNN Money*, 2014.

Gomez Rodriguez, M., Leskovec, J., and Krause, A. Inferring networks of diffusion and influence. In *SIGKDD*. ACM, 2010.

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

Jog, V. and Loh, P. Analysis of centrality in sublinear preferential attachment trees via the cmj branching process. *arXiv preprint arXiv:1601.06448*, 2016.

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

Khim, J. and Loh, PL. Confidence sets for the source of a diffusion in regular trees. *arXiv preprint arXiv:1510.05461*, 2015.

Koehler, J.O. *STASI: The untold story of the East German secret police*. Basic Books, 1999.

Luo, Wuqiong, Tay, Wee Peng, and Leng, Mei. How to identify an infection source with limited observations. *JSTSP*, 2014.

Milling, C., Caramanis, C., Mannor, S., and Shakkottai, S. Network forensics: Random infection vs spreading epidemic. In *SIGMETRICS*. ACM, 2012a.

Milling, C., Caramanis, C., Mannor, S., and Shakkottai, S. On identifying the causative network of an epidemic. In *Allerton Conference*, 2012b.

Milling, C., Caramanis, C., Mannor, S., and Shakkottai, S. Detecting epidemics using highly noisy data. In *MobiHoc*, 2013.

Pinto, P. C., Thiran, P., and Vetterli, M. Locating the source of diffusion in large-scale networks. *Physical review letters*, 109(6):068702, 2012.

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

Shah, D. and Zaman, T. Rumors in a network: Who's the culprit? *Information Theory, IEEE Transactions on*, 57 (8):5163–5181, Aug 2011.

Smith, A. 6 new facts about Facebook. *Pew Research*, 2014.

Viswanath, B., Mislove, A., Cha, M., and Gummadi, K.P. On the evolution of user interaction in facebook. In *Proc. of SIGCOMM WOSN*. ACM, August 2009.

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

Zhu, Kai and Ying, Lei. A robust information source estimator with sparse observations. *Computational Social Networks*, 2014.

Zhu, Kai, Chen, Zhen, and Ying, Lei. Locating the contagion source in networks with partial timestamps. *Data Mining and Knowledge Discovery*, 2015.

Supplementary material: Metadata-conscious anonymous messaging

A. Warm-up Example: Line Graph

We begin by considering the special contact network of a line graph. This example highlights how severely metadata can hurt anonymity; nonetheless, Section 3 illustrates that our seemingly-negative result on lines does not extend to higher-degree trees.

Consider a line graph $G(V, E)$ in which $V = \{0, 1, \ldots, n, n+1\}$, nodes $s_1 = 0$ and $s_2 = n+1$ are spies, and $E = \{(i, i+1) \mid i \in \{0, \ldots, n\}\}$. One of the n nodes between the spies is chosen uniformly at random as a source, denoted by $v^* \in \{1, \ldots, n\}$. When the message reaches a spy s_i, the spy collects at least two pieces of metadata: the timestamp T_{s_i} and the parent node p_{s_i} that relayed the message. 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.

In this section, we first show that under standard diffusion, the probability of source detection scales as $1/\sqrt{n}$. We also show that if spy nodes observed only timestamps and parent nodes, adaptive diffusion would achieve the optimal detection probability of $1/n$. However, adaptive diffusion passes extra metadata, which we call a *control packet*, to coordinate the message spread (details below). Control packets allow a spy to identify the source with probability 1. To overcome this challenge, we propose a new implementation of adaptive diffusion that provably achieves $1/\sqrt{n}$ (Proposition A.1). It is an open question if a smaller probability of detection can be achieved on a line.

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 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 5 (bottom) 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).

Adaptive diffusion on a line. Adaptive diffusion introduced in (Fanti et al., 2015) on a line is a random message spreading model governed by the location of a *virtual source* v_t at any (even) time t. At time 0, the source determines either the left or the right neighbor to be the next virtual source with equal probability. The message is propagated to the chosen node at time $t = 1$. At $t = 2$, the new virtual source v_2 propagates the message to its uninfected neighbor. At this point, three nodes are infected, with the virtual source v_2 at the center. At any given even time t, the infected subgraph is a subset of $t + 1$ nodes, centered around the virtual source v_t. At each even time t, the protocol has two options: keep the virtual source where it is, or pass it to the only neighbor who has not yet been a virtual source. 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. In the next two time steps, the message spreads in such a way that two more nodes are infected, and the virtual source is again at the center of the infected subgraph. This choice of virtual-

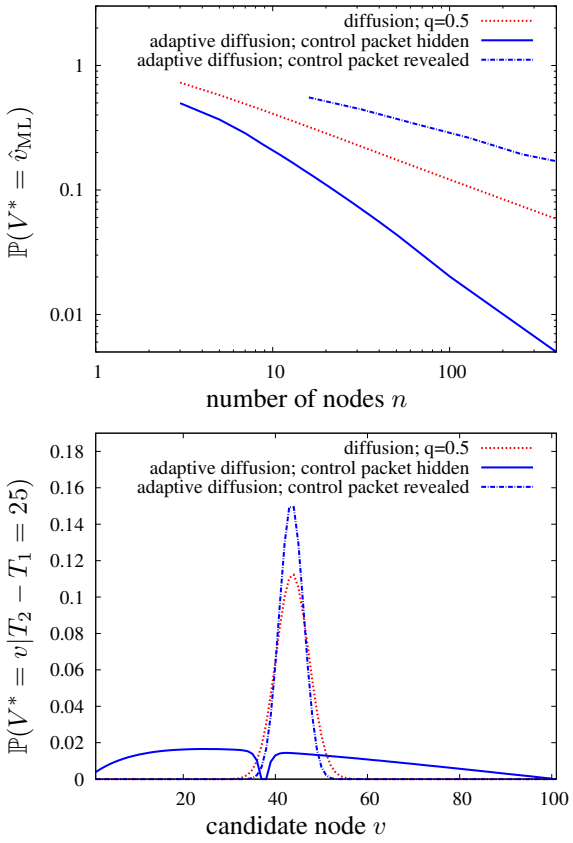

Figure 5. Comparisons of probability of detection as a function of n (top) and the posterior distribution of the source for an example with $n = 101$ and $T_2 - T_1 = 25$ (bottom). The line with 'control packet revealed' uses the Pólya's urn implementation.

source-spreading probability is optimal against a snapshot adversary, guaranteeing perfect obfuscation of the source.

Suppose spy nodes only observed timestamps and parent nodes but *not* control packets. The adversary could then numerically compute the ML estimate $\hat{v}_{\mathrm{ML}} = \arg\max_{v\in[n]} \mathbb{P}_{T_1-T_2|V^*}(T_{s_1} - T_{s_2}|v)$. Figure 5 shows the posterior is close to uniform (bottom) and the probability detection would scale as $1/n$ (top), which is the best one can hope for. Of course, spies *do* observe control packets, including the information to generate the randomness. This reveals the distance to the true source $\delta_H(v_T, v^*)$, and the true source is exactly identified with probability 1. We therefore introduce a new implementation (tailored for the line graph) that is robust to control packet information.

Adaptive diffusion via Pólya's urn. The random process governing the virtual source's propagation is identical to a Pólya's urn process (Johnson & Kotz, 1977). 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 $1-q$. The Bayesian interpretation of Pólya's urn processes shows that this process is equivalent to 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 A.1 of the Supplementary Material. Precisely, we provide an upper bound on the detection probability for such an adversary.

Proposition A.1. *When the source is uniformly chosen from n nodes between two spy nodes, the ML estimator achieves a detection probability upper bounded by*

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

Equipped with the ML estimator, we can also simulate adaptive diffusion on a line. Figure 5 (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.

A.1. Proof of Proposition A.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 follows 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 notation, 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 learn the time stamp T_1 with respect to the start of the infection. Further, once the adversary learns 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

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

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

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 (2)), it can only stay (the second equation in (3)), or both cases are possible (the second equation in (2)).

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 $Z \sim \mathrm{Binom}((t_1/2) - 2, q)$. The adversary can compute the

ML estimate:

$$\hat{v}_{\mathrm{ML}} = \begin{cases} \frac{T_1+2}{2} + \left\lfloor q\left(\frac{T_1-2}{2}\right)\right\rfloor & \text{if } T_1 \text{ even \& } D=\ell\,, \\ \frac{T_1+3}{2} + \left\lfloor q\left(\frac{T_1-1}{2}\right)\right\rfloor & \text{if } T_1 \text{ odd \& } D=\ell\,, \\ 1 + \left\lfloor (1-q)\left(\frac{T_1-1}{2}\right)\right\rfloor & \text{if } T_1 \text{ odd \& } D=r\,. \end{cases} \tag{4}$$

Together with the likelihoods in Eqs. (2) and (3), this gives

$$\mathbb{P}^{(\text{adaptive})}_{T_1,D|V^*,Q}\big(t_1, r, \hat{v}_{\mathrm{ML}} = v^* \big| v^*, q\big) = \frac{1}{2}(1-q)\, B\Big(\frac{t_1-1}{2} - v^*, \frac{t_1-3}{2}, q\Big) \mathbb{I}_{(\hat{v}_{\mathrm{ML}}=v^*)}\, \mathbb{I}_{(t_1 \text{ is odd})} \tag{5}$$

$$\mathbb{P}^{(\text{adaptive})}_{T_1,D|Q}\big(t_1, r, V^* = \hat{v}_{\mathrm{ML}} \big| q\big) = \frac{1}{2\,n}(1-q)\, B\Big(\frac{t_1-1}{2} - \hat{v}_{\mathrm{ML}}, \frac{t_1-3}{2}, q\Big)\, \mathbb{I}_{(t_1 \text{ is odd})} \tag{6}$$

$$\leq \frac{(1-q)}{2\,n}\Big(\frac{\sqrt{2}\,\mathbb{I}_{(t_1 \text{ is odd and } t_1>3)}}{\sqrt{\frac{t_1-3}{2}q(1-q)}} + \mathbb{I}_{(t_1=3)}\Big) \tag{7}$$

where $\hat{v}_{\mathrm{ML}} = \hat{v}_{\mathrm{ML}}(t_1, q, r)$ is provided in (4), 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 $2\times 0.4748/\sqrt{kq(1-q)}$), 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

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

Similarly, we can show that

$$\mathbb{P}\big(D=\ell, V^* = \hat{v}_{\mathrm{ML}}, T_1 \text{ is odd} \big| Q=q\big) \leq \frac{\sqrt{2}}{2\,n\,\sqrt{q(1-q)}}\sqrt{8\left\lfloor \frac{n-1}{2}\right\rfloor} + \frac{1}{n}\,, \tag{9}$$

$$\mathbb{P}\big(V^* = \hat{v}_{\mathrm{ML}}, T_1 \text{ is even} \big| Q=q\big) \leq \frac{q\sqrt{2}}{2\,n\,\sqrt{q(1-q)}}\sqrt{8\left\lfloor \frac{n}{2}\right\rfloor} + \frac{1+q}{2\,n}\,, \tag{10}$$

Summing up,

$$\mathbb{P}(V^* = \hat{v}_{\mathrm{ML}} | Q=q) \quad \leq \quad \sqrt{\frac{8}{n\,q\,(1-q)}} + \frac{2}{n}\,. \tag{11}$$

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

$$\mathbb{P}(V^* = \hat{v}_{\mathrm{ML}}) \quad \leq \quad \pi\sqrt{\frac{8}{n}} + \frac{2}{n}\,, \tag{12}$$

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

B. Regular Tree Analysis

B.1. Proof of Theorem 1

We begin by expanding some points regarding the ML estimator in Algorithm 2 that were omitted in Section 3. First, note that it is possible to derive an ML estimate without requiring the presence of a spine spy; the estimator described here uses a spine spy purely for ease of exposition. The omitted details are: (1) given a spy node s, how does the estimator find that spy node's pivot ℓ_s? (2) Why does timing information enable the estimator to disregard any subtree neighboring ℓ_{min} that contains at least one spy?

To answer the first question, consider the first spine spy s_0 and all spies in the feasible subtree. For each spy s in the feasible subtree (none of which lies on the spine), there exists a unique path between s and s_0. There exists a unique node on this path that is both part on the spine and closer to the true source than any other node in the path—this is precisely the pivot node. The estimator uses the observed metadata to infer the pivot, as well as its level in the infected subtree, for each spy in the feasible subtree. This inference proceeds by solving a system of equations:

$$h_{s,\ell_s} + h_{\ell_s,s_0} = |\mathcal{P}(s,s_o)|$$
$$h_{\ell_s,s_0} - h_{s,\ell_s} = T_{s_0} - T_s$$

where $\mathcal{P}(s,s_o)$ denotes the path between s and s_o, $h_{s,\ell_s} = \delta_H(s,\ell_s)$ denotes the distance from spy s_i to the pivot node ℓ_s, and h_{ℓ_s,s_0} is equal to $\delta_H(\ell_s,s_o)$ by construction. This system of equations always has a unique solution; hence the uniqueness of ℓ_s given s and s_0. The first equation holds by construction. The second equation holds because conditioned on the time at which the pivot receives the message T_{ℓ_s}, s_0 receives the message at time $T_{\ell_s} + h_{\ell_s,s_0}$, and s receives it at $T_{\ell_s} + h_{s,\ell_s}$.

Let L denote the set of pivots corresponding to each spy in the feasible subtree; in the example in Figure 2, $L=\{1,2\}$. Define $\ell_{min} = \text{argmin}_{\ell\in L} m_\ell$. That is, ℓ_{min} denotes the pivot closest to the true source in hop distance, i.e., whose level is lowest. Now consider the subtrees of depth $m_{\ell_{min}} - 1$ rooted at the neighbors of ℓ_{min}. The subtree including s_0 cannot contain the true source because we know the message traveled from ℓ_{min} to s_0. The source must therefore lie in one of the remaining $d-1$ neighbor subtrees, which we refer to as *candidate subtrees*.

We now argue that the estimator can rule out any candidate subtree of ℓ_{min} that contains at least one spy node. Suppose otherwise: there is a candidate subtree containing a spy s, and the source v^* is contained in that subtree. Then the path $\mathcal{P}(v^*,s)$ cannot pass through ℓ_{min} because ℓ_{min} does not belong to any of its own neighboring subtrees by construction. Then there must exist some node ℓ' on the spine such that $|\mathcal{P}(\ell',s)| < |\mathcal{P}(\ell_{min},s)|$. But this is a contradiction

because ℓ_{min} is chosen as the minimum-level pivot across all spies, and each spy has a unique pivot on the spine.

Since we can now rule out candidate subtrees with at least one spy, let $X+1$, $X \in \mathbb{N}$ be the number of candidate subtrees containing no spies. We use this notation because there will always be at least one candidate subtree with no spies (the one containing the true source). In Figure 2, $X = 0$. Thus, the ML estimator chooses one of the leaves in the remaining $X+1$ candidate subtrees uniformly at random. All remaining nodes in $V \backslash U$ have likelihood 0.

Probability of Detection: We condition on the lowest-level pivot node, ℓ_{min}, giving $\mathbb{P}(\hat{v}_{\text{ML}} = v^*) = \sum_{\ell_{min}} \mathbb{P}(\hat{v}_{\text{ML}} = v^* | \ell_{min}) \mathbb{P}(\ell_{min})$. Since ℓ_{min} lies on the spine, this is equivalent to conditioning on the distance of ℓ_{min} from the true source.

$$\mathbb{P}(\hat{v}_{\text{ML}} = v^*) = \sum_{k=1}^{\infty} \underbrace{\frac{(1-p)^{(|T_{d,k}|-1)}\, p}{|\partial T_{d,k}|}}_{\ell_{min}\ (k^{th}\text{ spine node) is a spy}} + \underbrace{(1-p)^{|T_{d,k}|} \mathbb{E}_X\Big[\frac{\mathbb{I}(X \neq d-2)}{(X+1)\,|\partial T_{d,k}|}\Big]}_{\ell_{min}\ (k^{th}\text{ spine node) not a spy}} \tag{13}$$

where $X \sim \text{Binom}(d-2, (1-p)^{|T_{d,k}|})$, $|T_{d,k}| = \frac{(d-1)^k - 1}{d-2}$ 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. Let $w = (1-p)$. We have that

$$\begin{aligned}\mathbb{E}_X\Big[\frac{\mathbb{I}(X \neq d-2)}{(X+1)\,|\partial T_{d,k}|}\Big] &= \frac{1}{|\partial T_{d,k}|}\Big(\mathbb{E}_X\Big[\frac{1}{X+1}\Big] - \frac{1}{d-1} w^{|T_{d,k}|\cdot(d-2)}\Big) \\ &= \frac{1}{|\partial T_{d,k}|}\Big(\frac{1}{(d-1)w^{|T_{d,k}|}}(1-(1-w^{|T_{d,k}|})^{d-1}) - \frac{1}{d-1} w^{|T_{d,k}|\cdot(d-2)}\Big)\end{aligned}$$

where the last line results from the expression for the expectation of $1/(1+X)$ when X is binomially-distributed. Namely if $X \sim \text{Binom}(\tilde{n}, \tilde{p})$, then $\mathbb{E}[1/(X+1)] = \frac{1}{(\tilde{n}+1)\tilde{p}}(1-(1-\tilde{p})^{\tilde{n}+1})$. Simplifying gives

$$\begin{aligned} P_D &= \sum_{k=1}^{\infty} \frac{1}{(d-1)^k}\Big[(d-1)pw^{|T_{d,k}|-1} + 1 - w^{|T_{d,k}|\cdot(d-1)} \\ &\quad -(1-w^{|T_{d,k}|})^{d-1}\Big] \\ &= p + \frac{1}{d-2} + \sum_{k=1}^{\infty} \frac{1}{(d-1)^k}\Big[pw^{|T_{d,k+1}|-1} - \\ &\quad w^{|T_{d,k}|\cdot(d-1)} - (1-w^{|T_{d,k}|})^{d-1}\Big] \\ &= p + \frac{1}{d-2} - \sum_{k=1}^{\infty} \frac{1}{(d-1)^k}\Big[w^{|T_{d,k+1}|} + \\ &\quad (1-w^{|T_{d,k}|})^{d-1}\Big]. \end{aligned}$$

where the last line holds because $|T_{d,k+1}| - 1 = |T_{d,k}| \cdot (d-1)$.

Expected hop distance: In the main paper, we lower bounded the expected hop distance by assuming that the estimator guesses the source exactly whenever (a) the pivot ℓ_{min} is a spy node or (b) the estimator chooses the correct candidate subtree. Therefore, if the pivot ℓ_{min} is at level k, we only consider estimates that are exactly $2k$ hops away. The estimator chooses an incorrect candidate subtree with probability $X/(X+1)$.

$$\mathbb{E}[\delta_H(\hat{v}_{\text{ML}}, v^*)] \geq \sum_{k=1}^{\infty} 2k(1-p)^{|T_{d,k}|}\mathbb{E}_{X_k}\Big[\frac{X_k \cdot \mathbb{I}(X_k \neq d-2)}{(X_k+1)}\Big]. \tag{14}$$

If $X_k \sim \text{Binom}(\tilde{n}, \tilde{p})$, where $\tilde{n}$ and $\tilde{p}$ depend on d and k, we have

$$\mathbb{E}_{X_k}\Big[\frac{X_k \cdot \mathbb{I}(X_k \neq \tilde{n})}{(X_k+1)}\Big] = \frac{1}{(\tilde{n}+1)\tilde{p}}\left[(1-\tilde{p})^{\tilde{n}} + \tilde{p}(1-(1-\tilde{p})^{\tilde{n}} + \tilde{n}) - 1 - \tilde{n}\tilde{p}^{\tilde{n}+1}\right]$$

Simplifying and substituting $\tilde{p} = (1-p)^{|T_{d,k}|}$ and $\tilde{n} = d-2$ gives the expression in the theorem.

Note that this bound is trivially 0 for $d = 3$, since we ignore all nodes in the correct candidate subtree; when $d = 3$, the source's candidate subtree is the only valid option if ℓ_{min} is not a spy. However, for a fixed p with $d \to \infty$, this lower bound approaches the upper bound of $2(1-p)$.

Obtaining a tighter bound is straightforward, but increases the complexity of the expression. These tighter bounds were used for the plots in the main paper. A tighter bound results from considering the cases when (a) the pivot ℓ_{min} is a spy node or (b) the estimator chooses the correct candidate subtree. In both cases, we ignore all but the most

distant estimates. For instance, if ℓ_{min} is on the spine at level k, then the estimate will be at most $2(k-1)$ hops away. Using this rule for both cases (a), we compute the probability of selecting one of the most distant options:

$$a_k \equiv \frac{d-2}{d-1}(1-p)^{|T_{d,k}|(d-1)}$$

and for case (b):

$$b_k \equiv p\frac{d-2}{d-1}(1-p)^{|T_{d,k}|-1}$$

Overall, we get a lower bound of

$$\mathbb{E}[\delta_H(\hat{v}_{\mathrm{ML}}, v^*)] \geq \sum_{k=1}^{\infty} 2(kr_k + (k-1)(a_k+b_k))$$

C. Irregular Tree Analysis

C.1. Proof of Proposition 4.1

All nodes in $V \backslash U$ have likelihood zero, as discussed in the proof of Theorem 1 (recall that V denotes the set of all nodes, and U denotes the set of candidate nodes). The only randomness in adaptive diffusion spreading occurs when a spine node with uninfected neighbors decides which of its neighbors will be added to the spine next. Thus, the (log) likelihood of a candidate source is the sum of the (log) likelihoods of all candidate spine nodes starting at the candidate source. Regardless of which node $u \in U$ is the true source, the spine must pass through ℓ_{min}; since there is a unique path between u and ℓ_{min} over trees, the only feasible sequence of candidate spine nodes starting at candidate u must traverse $\mathcal{P}(u, \ell_{min})$. By the Markov property of the adaptive diffusion spreading mechanism, we only need to consider the likelihood of a candidate spine prior to reaching ℓ_{min}. The propagation of the spine thereafter is conditionally independent of the true source, and therefore equally-likely for all candidates. The maximum likelihood estimator must therefore compute the likelihood of each such candidate sub-spine $\mathcal{P}(u, \ell_{min})$. Since each spine node v chooses one of its uninfected neighbors uniformly at random to be the next spine node, the choice of next spine node is simply $1/(\deg(v)-1)$. Similarly, the likelihood of candidate source u sending the spine in a particular direction is $1/\deg(u)$. The overall likelihood of a candidate is therefore proportional to the product of these degree terms.

C.2. Analysis of spy+snapshot adversarial model

We follow closely the proof of Theorem 1 in Appendix B.1. Given a snapshot at a certain even time T, if there are at least two spy nodes infected at time T, then the adversary can perform the exact same estimation as he did with only spy nodes with $T \to \infty$. We only need to carefully analyze what happens when there are only one spy infected or no spies are infected.

We condition on the lowest-level pivot node, ℓ_{min}, giving $\mathbb{P}(\hat{v}_{\mathrm{ML}} = v^*) = \sum_{\ell_{min}} \mathbb{P}(\hat{v}_{\mathrm{ML}} = v^* | \ell_{min}) \mathbb{P}(\ell_{min})$. Since ℓ_{min} lies on the spine, this is equivalent to conditioning on the distance of ℓ_{min} from the true source. We first define $|S_{d,T}| = 1 + d((d-1)^{T/2}-1)/(d-2)$ as the number of nodes infected at time T, and $|\partial S_{d,T}| = d(d-1)^{(T/2)-1}$ as the number of leaves in the infected subtree. Then,

$$\begin{aligned}
\mathbb{P}(\hat{v}_{\mathrm{ML}} = v^*) \;=\; & \underbrace{\frac{(1-p)^{|S_{d,T}|-1}}{|\partial S_{d,T}|}}_{\text{no spy}} + \\
& \sum_{k=1}^{T/2} \Bigg\{ \underbrace{\frac{(1-p)^{(|T_{d,k}|-1)}\, p}{|\partial T_{d,k}|}}_{\ell_{min}\ (k^{th}\text{ spine node) is a spy}} + \\
& \underbrace{(1-p)^{|T_{d,k}|}(1-(1-p)^{|S_{d,T}|-|T_{d,k+1}|})\mathbb{E}_X\Big[\frac{\mathbb{I}(X \neq d-2)}{(X+1)\,|\partial T_{d,k}|}\Big]}_{\ell_{min}\ (k^{th}\text{ spine node) not a spy}} + \\
& \underbrace{(1-p)^{|S_{d,T}|-(|T_{d,k+1}|-|T_{d,k}|)}\mathbb{E}_X\Big[\frac{\mathbb{I}(X \neq d-2)}{|\partial S_{d,T}| - (d-2-X)|\partial T_{d,k}|}\Big]}_{\text{all spy descendants of } k\text{-th spine node}} \Bigg\},
\end{aligned} \tag{15}$$

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.

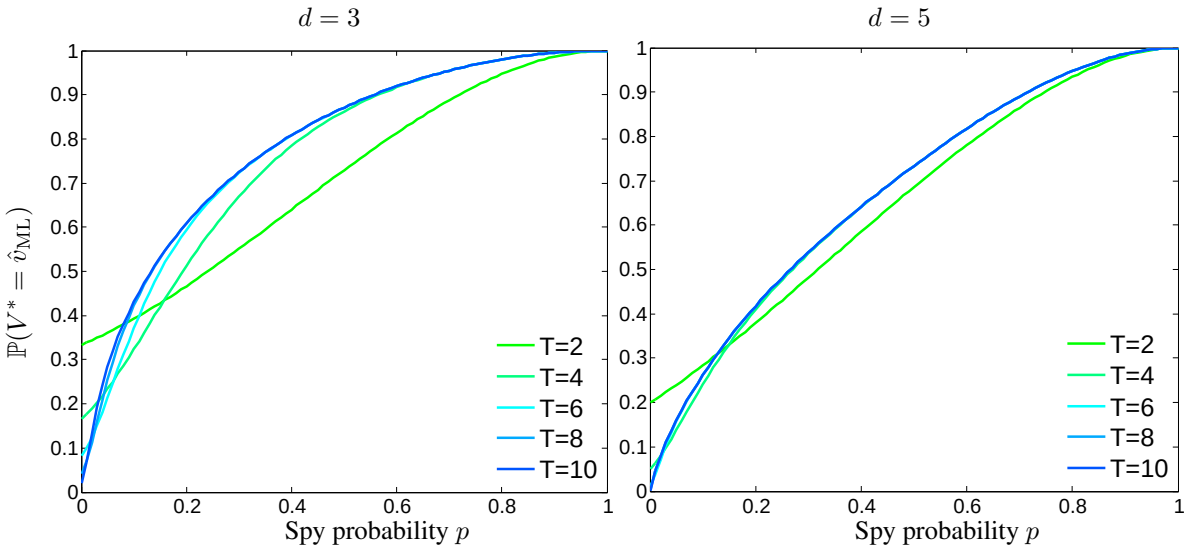

Figure 6. Probability of detection under the spies+snapshot adversarial model. As estimation time and tree degree increase, the effect of the snapshot on detection probability vanishes.