How Can Randomized Routing Protocols Hide Flow Information in Wireless Networks?

Shangqing Zhao, *Student Member, IEEE,* Zhuo Lu, *Member, IEEE,* and Cliff Wang, *Fellow, IEEE*

***Abstract*—Preventing the source-destination network flow information from being disclosed is pivotal for anonymous wireless network applications. However, the advance of network inference, which is able to obtain the flow information without directly measuring it, poses severe challenges towards this goal. Randomized routing is capable of hiding the flow information by injecting substantial errors to the network inference process. In this paper, we systematically study the behavior of randomized routing protocols, and categorize them into three templates, k-random-relay, k-random-neighbor and k-random-path based on their routing behaviors. We propose technical models to characterize these templates in terms of their induced inference errors and their delay costs. We also use simulations to validate the theoretical results. Our work provides the first systematic study on understanding both the benefit and the cost of using randomized routing to hide the flow information in wireless networks.**

***Index Terms*—Network inference and tomography; random routing; security; random walk.**

I. Introduction

In a wireless network, network flow information (i.e., the flow data rates of source-destination pairs on end-to-end paths) is the essential knowledge about the network. Equipped with such knowledge, malicious adversaries will know who is communicating with whom in the network or how much data rate a pair of communicating parties has, and then launch powerful, effective attacks targeting the network [1], [2]. For example, in military wireless networks, one node associated with a lot of traffic-intensive flows may be the master node which often releases critical decisions. Accordingly, the adversary can launch powerful attacks by compromising this master node. Furthermore, a consecutive high-rate flow can infer that the relationship of two end-nodes are close, which can be leveraged by attackers for network partition. In many large-scale wireless networks, such as wireless sensor network (WSN) [3] or mobile ad-hoc network (MANET) [4], directly observing the end-to-end flow information is not always possible or even infeasible because of the prohibition in anonymous networks or the measurement traffic overhead [5]–[12]. To acquire network flow information, adversaries can leverage the method of *network inference* [6], [13]–[16] to infer the end-to-end flow rates through eavesdropping on wireless link activities, which is widely feasible in wireless networks because of the broadcast nature of the wireless medium.

Network inference was originally designed for the effective network management and diagnosis, therefore most existing studies focus on optimizing the inference performance [6], [15]–[20]. However, from the adversary's perspective, such a line of work indeed exposes the vulnerability of leakage of network flow information, in which the adversary can simply obtain such information through wireless link measurements.

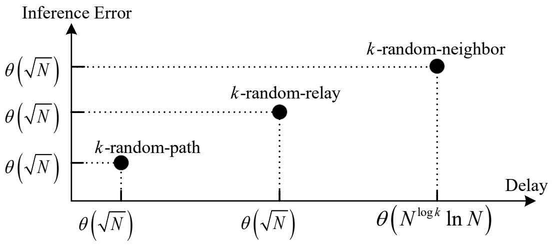
How accurately the adversary can obtain the flow information depends on the inference method the adversary uses, network traffic patterns, and routing protocols. Although it has been noted recently [21] that the inference error for an adversary can be maximized if a network can make sure the adversary has no knowledge of how packets are routed in the network, little progress has been made to reveal how a network can indeed meet this goal. Obviously, a larger inference error helps a network to hide its flow information against leakage more. To achieve this, the network can make the routing of packets unpredictable to adversaries. For example, if a deterministic routing protocol like the shortest path routing [22] is used, it can be very likely that the adversary can accurately infer the flow information from link measurements based on standard network inference methods [23]. On the contrary, if a routing protocol is randomized, such as using Tor [10], the intermediate relay nodes are randomly selected to form an end-to-end path, yielding unpredictable behavior of packet forwarding observed by an outside adversary.

In this paper, we study the security benefits of randomized routing protocols against malicious network inference as well as their associated costs. In particular, we focus on analyzing three major templates for randomized protocols based on common routing behaviors in wireless networks.

1) k-random-relay: k forwarding nodes (called relays) are selected randomly in a network. The routing path for each packet must contain such relays.
2) k-random-neighbor: in which the next hop is selected randomly. Specifically, for each packet, at each hop, one neighbor is selected randomly from k neighbors with the shortest distances (the distance is measured by the number of hops) to the destination.
3) k-random-path: each packet is sent by one randomly selected path from the k shortest end-to-end paths.

These templates capture the majority of randomized behaviors during packet forwarding that can make the entire routing unpredictable. In this paper, we use an asymptotic approach to theoretically model these routing templates and measure their induced inference errors. We also analyze the incurred delay cost of three templates. Our major results can be summarized

Shangqing Zhao and Zhuo Lu are with Department of Electrical Engineering, University of South Florida, Tampa FL, 33620.
Emails: {zhuolu@, shangqing@}usf.edu.
Cliff Wang is with Department of Electrical and Computer Engineering with North Carolina State University, Raleigh NC, 27695.
Email: cliffwang@ncsu.edu.



(a)

	Inference Error	Delay
k-random-relay	$\theta\left(\frac{\sqrt{N}}{(k+2)}\right)$	$\theta\left((k+1)\sqrt{N}\right)$
k-random-neighbor	$\theta\left(\frac{N}{(\ln N)^{-1}+\sqrt{N}}\right)$	$\theta\left(N^{\log k}\ln N\right)$
k-random-path	$\theta\left(\frac{\sqrt{N}(k-1)^2}{2k^2}\right)$	$\theta\left(\sqrt{N}\right)$

(b)

Fig. 1. Asymptotic inference error and delay performance for a network with N nodes and $\sqrt{N}$ flow, where (a) demonstrates the relationship of them, and (b) provide more exact results.

in Fig. 1, which depicts the lower bound of inference errors for three templates scaled by the delay cost in asymptotic notations for a network with N nodes and $\sqrt{N}$ active end-to-end flows under the condition that N is sufficiently large. From Fig. 1(a), we see that the inference errors induced by all templates are on the same order of $\sqrt{N}$ with constant difference. Both k-random-relay and k-random-path templates have delay costs on the same order of $\sqrt{N}$. The k-random-neighbor template incurs a delay cost of $N^{\log k}\ln N$. Our results indicate that k-random-path achieves the inference error with the same order of the other two templates while maintaining the lowest delay cost. But it requires knowing the global path information of the network, and in practice, such information is generally unavailable or prohibited to know. For a large k, the delay of k-random-neighbor is significantly larger than others, but the inference error is still on the same order of the others. As a result, k-random-neighbor with a large k should be avoided. From Fig. 1(b), for k-random-relay, both the inference error and the delay are larger than k-random-path by a constant order of magnitude. In addition, k-random-relay does not require to know the global information, thus a number of real-world applications (e.g., Onion routing or Tor [10]) leverage this template to achieve the anonymity.

Our main contributions are summarized as follows.

- We are the first to study the vulnerability of the leakage of network flow information from the routing protocol perspective, and reveal that randomized routing protocols help to prevent revealing flow information by inflicting large inference errors. Whereas, a large inference error is always associated with a large delay cost.
- We propose three templates based on routing behavior, i.e., k-random-relay, k-random-neighbor and k-random-path. Then we systematically characterize and model these templates, and investigate the inference error as well as the incurred delay cost of each template. Our theoretical results verify that each randomized template is able to hide the flow information with different capabilities.
- We conduct comprehensive simulations to evaluate the inference error and delay of each template under a practical network inference setup. Experimental results confirm the relationship between the inference error and the delay cost.

Our paper explores the fundamental reason why randomized routing strategies can prevent the information leakage against network inference. The results from this paper can not only be used to show and compare the difference among routing templates, but also provide a benchmark or guideline when designing new randomized routing strategies.

II. Preliminary and Problem Statement

In this section, we first present the network model and the preliminary of network inference. Then, we state our research problems. The notations of this paper are summarized as follows. (i) $f(n) = O(g(n))$ denotes there exists a constant c such that $f(n) \leq cg(n)$; $f(n) = \Omega(g(n))$ means $g(n) = O(f(n))$; $f(n) = \Theta(g(n))$ means $f(n) = O(g(n))$ and $f(n) = \Omega(g(n))$. (ii) Given a $m \times n$ matrix $\mathbf{A}$ with entry x_{ij}, then $\|\mathbf{A}\|_F = \sqrt{\sum_{i=1}^{m}\sum_{j=1}^{n} x_{ij}^2}$, and $\mathbf{tr}\{\mathbf{A}\}$ is the trace of $\mathbf{A}$. (iii) For a vector $\mathbf{v} = [v_1, \cdots, v_n]$, $\|\mathbf{v}\|_2 = \sqrt{\sum_{i=1}^{n} v_i^2}$; $|\cdot|$ denotes the cardinality operator.

A. Network Model

We model the wireless network by using a random geometric graph (RGG) [24] denoted by $\mathcal{G} = (\mathcal{V}, \mathcal{L})$, where $\mathcal{V}$ is the node set and $\mathcal{L}$ is the undirected link set. Let $N = |\mathcal{V}|$ and $L = |\mathcal{L}|$. In this network, N nodes are randomly placed in a region $\Omega = [0, \sqrt{N/\lambda}]^2$, where λ denotes the node density. We assume that λ is sufficiently large such that the network is connected asymptotically almost surely [24]. Denote by r the transmission range of each node.

In the network, packet exchange occurs in node pairs, yielding multiple end-to-end data flows. We denote by $\mathcal{F}$ the end-to-end flow set consisting of the potential flow for each node pair. Obviously, $|\mathcal{F}| = N(N-1)/2$, which is the number of node pairs. Each flow $f_i \in \mathcal{F}$ is associated with a metric x_i denoting the data rate on flow f_i. Denoted by a column vector $\mathbf{x} = [x_i]_{i\in[1,|\mathcal{F}|]}$ the flow rate vector for the network. We consider $x_i = 0$ if flow f_i does not exist (i.e., there is no communication).

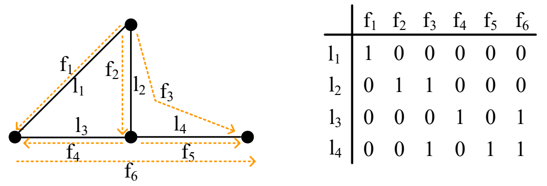
The flow rate vector $\mathbf{x}$ contains two types of information:

1) who is communicating with whom in the network;
2) how much data rate a pair of communicating parties has.

The disclosure of such information is undesirable or even prohibited in many network security scenarios [7], [25], [26].

B. Attack Model and Network Inference

What kind of methods an adversary can use to obtain $\mathbf{x}$? Note that $\mathbf{x}$ is not generally available to the adversary because flow information is indicated at network or higher layers [27], whose data is usually encrypted at the physical



	f1	f2	f3	f4	f5	f6
l1	1	0	0	0	0	0
l2	0	1	1	0	0	0
l3	0	0	0	1	0	1
l4	0	0	1	0	1	1

(a) Network topology. (b) A routing matrix $\mathbf{A}$.

Fig. 2. Example network topology (a) and the routing matrix (b), where the link set is $\mathcal{L} = \{l_1, \cdots, l_4\}$, and the flow set is $\mathcal{F} = \{f_1, \cdots, f_6\}$.

or link layer. Therefore, the adversary has to infer such information based on physical/link-layer activities, which is called network inference [18]. In this paper, we consider an omniscient adversary who (i) knows the network topology and (ii) knows the routing protocol used in the network, and (iii) is capable of monitoring each link $l_i \in \mathcal{L}$ in the network. This strong attack model enables us to clearly compare the benefits of different randomized routing protocols under the worst-case standard.

By letting column vector $\mathbf{y} = [y_1, y_2, \cdots, y_L]^T$ (where y_i denotes the rate of link l_i and $\cdot^T$ is the matrix transpose operator), the goal of the adversary is written as obtaining the estimated value of $\mathbf{x}$, denoted by $\hat{\mathbf{x}}$, from the measured link rate vector $\mathbf{y}$. Network inference is an approach to achieve the adversary's goal. In particular, the relationship between flow rate vector $\mathbf{x}$ and the link rate vector $\mathbf{y}$ can be captured by the following linear system.

$$\mathbf{y} = \mathbf{A}\mathbf{x}, \tag{1}$$

where $\mathbf{A}$ is the routing matrix with size $L \times |\mathcal{F}|$, whose entry

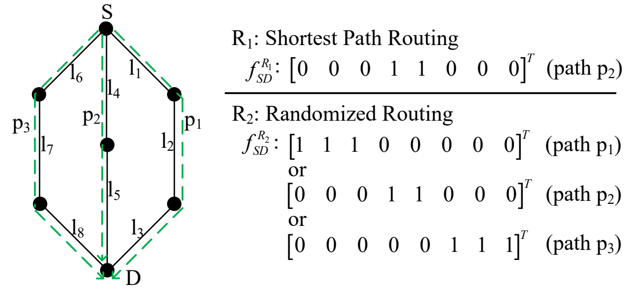
$$a_{ij} = \begin{cases} 1, & \text{if flow } f_j \text{ goes through link } l_i; \\ 0, & \text{otherwise.} \end{cases} \tag{2}$$

The routing matrix $\mathbf{A}$ demonstrates how a flow is concatenated by links, which is the function of routing protocols. Therefore, given a network, the routing matrix can be usually determined by the routing protocol. Fig. 2 shows an illustrative example of a network topology and its corresponding routing matrix $\mathbf{A}$. In this network, we have the link set $\mathcal{L} = \{l_1, \cdots, l_4\}$ and flow set $\mathcal{F} = \{f_1, \cdots, f_6\}$, where each flow $f_i \in \mathcal{F}$ is determined by the shortest path routing protocol. For example, flow f_1 only goes through link l_1 because this path, with length 1 (the length is measured by the number of hops), is the shortest one; and the entry corresponding to f_1 and l_1 is therefore 1 in the routing matrix $\mathbf{A}$ as shown in Fig. 2(b).

In order to estimate $\mathbf{x}$, the adversary must know the routing matrix $\mathbf{A}$ in (1), which is usually an under-determined system. If the adversary indeed knows such information, it can use standard network inference methods (e.g., $\mathcal{L}_1$-norm minimization [23]) to obtain $\hat{\mathbf{x}}$, the estimated version of $\mathbf{x}$. The inference error can be denoted as

$$I_R = \|\hat{\mathbf{x}} - \mathbf{x}\|_2. \tag{3}$$

For example, in Fig. 2, only flow f_3 has data steam with rate 10bps, i.e., $\mathbf{x} = [0, 0, 10, 0, 0, 0]$. Then the attacker can obtain the link rate vector $\mathbf{y} = [0, 10, 0, 10]$, indicating that links l_2 and l_4 have data transmission with rate 10bps. If the routing matrix $\mathbf{A}$ is also available, the attacker can obtain the estimated flow rate vector $\hat{\mathbf{x}}$. Note that we do not confine the adversary to any particular inference method. If we know what method the adversary uses, we may provide a method-specific defense. In this paper, we assume the worst case that the network inference approach used by the adversary is unknown to us.



R$_1$: Shortest Path Routing

$f_{SD}^{R_1}$: $[0\ \ 0\ \ 0\ \ 1\ \ 1\ \ 0\ \ 0\ \ 0]^T$ (path p$_2$)

R$_2$: Randomized Routing

$f_{SD}^{R_2}$: $[1\ \ 1\ \ 1\ \ 0\ \ 0\ \ 0\ \ 0\ \ 0]^T$ (path p$_1$)
or
$[0\ \ 0\ \ 0\ \ 1\ \ 1\ \ 0\ \ 0\ \ 0]^T$ (path p$_2$)
or
$[0\ \ 0\ \ 0\ \ 0\ \ 0\ \ 1\ \ 1\ \ 1]^T$ (path p$_3$)

(a) Network topology. (b) One column in $\mathbf{A}$ under protocols.

Fig. 3. The relationship between link and flow rates under routing protocols: R_1 = shortest path, and R_2 = random routing.

C. Randomized Routing Protocols: Benefit and Cost

In general, the routing matrix $\mathbf{A}$ is not immediately available to the adversary. However, as the adversary is assumed to know what routing protocol is used, it can build the routing matrix by itself given the network topology. If a deterministic routing protocol (e.g., shortest path) is used, it can be very likely that the adversary builds the routing matrix $\hat{\mathbf{A}}$ in close value to the true routing matrix $\mathbf{A}$, then obtains an accurate estimate of $\mathbf{x}$.

Randomized routing protocols make the behavior of routing packets unpredictable and accordingly incurs more inference error than deterministic ones to the adversary. To illustrate how a randomized routing protocol may work, we consider a simple network topology in Fig. 3(a). The network consists of 8 links (i.e., links $l_1 - l_8$) and one traffic flow f_{SD} between nodes S and D, which contains three potential routing paths (i.e., paths p_1, p_2 and p_3). Because there is only one flow, the routing matrix $\mathbf{A}$ only has one column representing the flow f_{SD}. We define a path set $\mathcal{P}_{SD} = \{p_1, p_2, p_3\}$ for flow f_{SD}, and two routing protocols R_1 and R_2 for the shortest path routing protocol and a randomized routing protocol, respectively.

Under protocol R_1, flow f_{SD} deterministically uses path p_2 to forward packets because it is the shortest path, leading to no randomness to choose a path. Aware of the shortest-path routing, the adversary can immediately reconstruct a routing matrix $\hat{\mathbf{A}}$ which is exactly the same as the ground-truth matrix $\mathbf{A}$. Then, the adversary can use it to accurately recover $\mathbf{x}$.

Under protocol R_2, a path is selected uniformly at random from the path set $\mathcal{P}_{SD}$. Then, the adversary's estimation $\hat{\mathbf{A}}$ equals to $\mathbf{A}$ with probability $1/3$, as shown in Fig. 3(b). This extra error due to the routing matrix mismatch will be introduced to the network inference, offering higher protection of the network flow information. Note that for this illustrative example in Fig. 3, the adversary may further determine which path is chosen by observing which links are active. However, simply observing which links are active cannot be applied to a multi-hop wireless network, where a large number of node pairs exchange packets simultaneously in the network.

On one hand, protocol R_2 causes more inference error to the adversary. On the other hand, however, R_2 incurs more delay during the data transmission because packets are not always be forwarded along the shortest path. Therefore, the cost of R_2 is the increase of delay. To provide formal models, we define the benefit (the inference error) and the cost (the delay) of randomized routing as follows.

Definition 1 (Inference Error and Delay): Within a network $\mathcal{G}$ with N nodes under a routing protocol R, the inference error for the adversary is measured by $I_R = \mathbb{E}\|\hat{\mathbf{x}} - \mathbf{x}\|_2$, where $\hat{\mathbf{x}}$ and $\mathbf{x}$ are the adversary's estimated flow rate vector and true flow rate vector, respectively. The delay between a node pair is the average number of hops for data delivery between the pair. Given a routing protocol R, the average delay h_R, is obtained by averaging the delays over all node pairs.

The adversary cannot directly observe the end-to-end flow information, and the flow information can be obtained only through inference. This is why we use the inference error as the benefit of different randomized routing strategies. The benefit does not confine to protecting the flow information. For example, it can also be applied to balance the traffic load or mitigate the wireless link failures. However, the goal of this paper is how to provide defense against network inference, and we find that randomized routing can achieve this goal. Therefore, we explore the randomized routing strategy from the security perspective, and characterize the benefit as the inference error, which is directly related to the defense performance.

In addition, we use the number of hops as the cost since it is directly related to the inference error. The number of hops is easily available and widely used information serving as the cost for routing discovery in many wireless network protocols, such as AODV [28] in MANET. Therefore, in our paper, we also use the number of hops as the cost metric to measure the performance of each template, and focus on analyzing protocols using the number of hops as the performance cost.

D. Templates for Randomized Routing and Problem Statement

To formally characterize randomized routing protocols, we propose three templates to analyze the benefit and cost of randomized routing in wireless networks.

1) T_1: k-random-relay: for each packet, the routing path is formed by selecting k nodes (called relays) uniformly at random from $\mathcal{V}$ excluding the source and destination nodes. Then, the packet is transmitted through these relays. The shortest path routing is used between two consecutive relays. Onion routing used in the Tor [10] or Crowds [11] networks are popular real-world applications of this template although they are currently used in wireline networks.
2) T_2: k-random-neighbor: for each packet, at each hop, one neighbor is selected with equal probability from k neighbors with shortest distances to the destination. The random grid routing [29] and greedy forwarding routing [30] are two simplistic versions of this template.

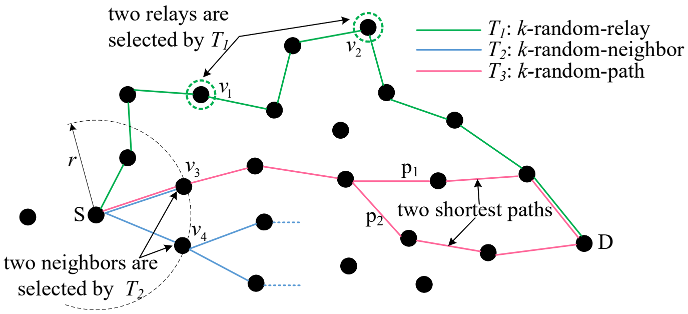


Fig. 4. Examples of k-random-relay, k-random-neighbor, and k-random-path routing templates for $k = 2$.

3) T_3: k-random-path: each packet is transmitted through one path selected uniformly at random from k shortest paths. This template is available when the overall path information is available. The template can be considered as a randomized version of the k-shortest-path routing protocol [22].

The examples for the three templates is shown in Fig. 4, which has one flow f_{SD} between nodes S and D with $k = 2$. Under k-random-relay, two relays $v_1, v_2 \in \mathcal{V}$ are randomly selected, then a packet can be transmitted through these two relays (i.e., the green line), where the paths between the source and the first relay, between the two relays, and between the second relay to the destination are all the shortest paths. Under k-shortest-neighbor, the source S randomly picks one neighbor v_4 from two neighbors v_3 and v_4 which have the shortest distance towards the destination D. Then, v_4 adopts the same rule for the second hop, and so on. The path for the k-random-path template is randomly selected from 2 shortest paths p_1 and p_2.

In these template, k is an independent variable of N and is determined by practical applications or the real-world scenarios. For example, in the Tor network, k is fixed to 3 no matter how many nodes in the network. In this paper, to provide randomness, we assume $k > 1$ is a constant independent of N. The proposed templates cover a wide range of randomized routing behaviors. However, which template is more suitable for a network and how to choose the routing template are based on several factors such as node density and communication rage of each node. The template selection for a particular network scenario is orthogonal to the research in this paper that focuses on performance and cost analysis. Our objective in this paper is to model and analyze the inference error and delay of each template to offer a fundamental basis or guideline for designing practical randomized routing protocols.

III. Theoretical Modeling and Strategy

In this section, we provide a theoretical metric to measure the inference error induced by randomized routing. Then, we discuss our strategy to model routing templates.

A. Genie Bound

It is expected that randomized routing increases the inference error I_R, which is related to a particular inference method used by the adversary to derive $\hat{\mathbf{x}}$ in (1). To remove such a dependency, we use the genie bound [31] to measure I_R as it is a generic metric regardless of the inference method. Specifically, for the under-determined linear system (1), the genie bound, serving as a lower error bound of all possible inference methods, can be obtained in three steps:

1) form a new deterministic linear system with the assistance of a genie, which is expressed as
$$\mathbf{y} = \mathbf{A}_g \mathbf{x}_g, \tag{4}$$
where $\mathbf{x}_g$ denotes the flow rate vector for node pairs that indeed have real traffic flows, obtained by removing zero entries (e.g., non-existing flows) from $\mathbf{x}$. $\mathbf{A}_g$ is the routing matrix presenting the relationship between real flows and links;
2) derive the least square estimation of $\mathbf{x}_g$ as
$$\hat{\mathbf{x}}_g = (\mathbf{A}_g^T \mathbf{A}_g)^{-1} \mathbf{A}_g^T \mathbf{y}; \tag{5}$$
3) obtain the genie bound by deriving the minimum mean square error between $\hat{\mathbf{x}}_g$ and $\mathbf{x}_g$, i.e.,
$$G(\mathbf{x}_g) = \mathbb{E}\left(\|\hat{\mathbf{x}}_g - \mathbf{x}_g\|_2^2\right). \tag{6}$$

Note that in step 1, with the help of the genie, the under-determined linear system (1) is converted into a determined system (4). This removes the effect of the choice of the inference method used by the attacker. The genie bound is a general and method-independent bound, and is widely used in solving under-determined system to provide a lower error bound for any inference method. Formally, we define the genie bound as a generic metric to quantify the inference error I_R as follows.

Definition 2 (Genie Bound Metric): Given the link rate vector $\mathbf{y}$, and routing matrix $\mathbf{A}$ determined by the routing template T, the inference error I_T can be rewritten as
$$I_T = \mathbb{E}\left(\|\hat{\mathbf{x}}_g - \mathbf{x}_g\|_2^2|\right), \tag{7}$$
where $\hat{\mathbf{x}}_g$ and $\mathbf{x}_g$ are derived in the three-step genie bound procedure.

B. Routing Template Modeling

The inference error is due to the randomness in selecting an end-to-end path for a flow. In a real network, it is very likely that there are multiple paths for a flow, and how to select a path is based on the routing protocol. For example, in Fig. 3, the shortest path protocol simply selects path p_2 whereas random routing selects one randomly from paths p_1, p_2 and p_3.

Fundamentally, these three templates provide different behaviors to select a path from the total path set of a flow. For example, in Fig. 3, under the template T_3, flow f_{SD} randomly selects a path from $\mathcal{P}_{SD}$ if it is available. However, if only the neighbor information is available and the template T_2 is used, T_2 will randomly choose a link from the link set $\{l_1, l_4, l_6\}$ for the next hop to forward packets, leading to a different manner to choose a path from f_{SD}. All these random factors eventually lead to a random path selection for a flow. Therefore, our strategy is to use a general path-selection model to capture all behaviors of different randomized routing templates.

Before modeling the difference among different routing templates, we first define $\mathcal{P}_i$ as the path set consisting of all potential paths for the flow $f_i \in \mathcal{F}_g$. Denoted by $\mathcal{P}_\Pi = \{\mathcal{P}_i\}_{i\in[1,F]}$ the path set for all source-destination pairs in the network. To build the relationship between a path and its corresponding vector in the routing matrix, we define a function mapping J which maps paths to their corresponding columns in the routing matrix. For example, in Fig. 3, $J(p_1) = [1,1,1,0,0,0,0,0]^T$.

Based on the previous definitions, we introduce three matrices as follows.

1) The overall routing matrix $\mathbf{M} = [\mathbf{m}_1, \cdots, \mathbf{m}_F]$ consisting of all potential paths. It means that each entry $\mathbf{m}_i = J(\mathcal{P}_i)$ for $1 \le i \le F$, and $\mathbf{M} = J(\mathcal{P}_\Pi)$.
2) The template matrix $\mathbf{T} = \text{diag}(\mathbf{t}_1, \cdots, \mathbf{t}_F) \in \mathbb{R}^{|\mathcal{P}_\Pi| \times \sum g_i}$ is a rectangular diagonal matrix. Each entry $\mathbf{t}_i$ is a $|\mathcal{P}_i| \times g_i$ matrix, where $1 \le g_i \le |\mathcal{P}_i|$ denotes the number of paths selected by the routing template for the flow f_i. Each column in $\mathbf{t}_i$ has only one entry with value 1 denoting which path is selected and the remaining entries are all zeros.
3) The selection matrix $\mathbf{C} = \text{diag}(\mathbf{c}_1, \cdots, \mathbf{c}_F) \in \mathbb{R}^{\sum g_i \times F}$, where each $\mathbf{c}_i$ is a $g_i \times 1$ all zero column vector except one 1 indicating which path is selected for the actual packet transmission.

In the following, by leveraging the above three matrices, we propose a technical model to separate different factors in performance modeling for randomized routing.

Model 1: [Routing Template] Given a network topology $\mathcal{G}$, and the routing template T, the routing matrix $\mathbf{A}_g$ can be modeled as $\mathbf{A}_g = \mathbf{MTC}$, where matrices $\mathbf{M}$, $\mathbf{T}$, and $\mathbf{C}$ are defined as above.

We use the network topology in Fig. 3(a) as an example. Assume the routing protocol is T_3: k-random-path with $k = 2$, and there is only one flow f_{SD} in the network. Then, the routing matrix $\mathbf{A}_g \in \mathbb{R}^{8\times 1}$ can be expressed as

$$\mathbf{A}_g = \underbrace{\begin{bmatrix} 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 \end{bmatrix}^T}_{\mathbf{M}} \times \underbrace{\begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 0 \end{bmatrix}}_{\mathbf{T}} \times \underbrace{\begin{bmatrix} 1 \\ 0 \end{bmatrix}}_{\mathbf{C}}. \tag{8}$$

Three columns in $\mathbf{M}$ denote all possible paths between nodes S and D. Thus, $\mathbf{M}$ only relies on the network topology and is independent of routing behavior. There are two columns in $\mathbf{T}$, meaning two possible path selections (i.e., p_1 and p_2) in k-random-path with $k = 2$. The last matrix $\mathbf{C}$ denotes the path p_1 is finally used. Hence, the routing behavior decides $\mathbf{T}$ and $\mathbf{C}$.

Remark 1: We decompose the routing matrix $\mathbf{A}_g$ into $\mathbf{MTC}$ such that each matrix represents one factor to affect the values in $\mathbf{A}_g$. The overall routing matrix $\mathbf{M}$ only depends on the network topology. The template matrix $\mathbf{T}$ relies on a randomized routing template, and which template to be used is based on many factors such as the link capacity constraints or wireless interference. The selection matrix $\mathbf{C}$ indicates which path is used to send packets. Each of its column vector has only one entry with value 1, meaning that only one path is selected for a source-destination pair. Note that this model does not consider the scenario that one packet is sent over multiple paths, which is not typical in wireless networks due to the broadcast nature of the wireless channel.

IV. Theoretical Results

In this section, we first present the theoretical results of the inference error and delay of each randomized routing template. Then, we provide the proofs of the results.

A. Main Results

1) Inference Error: We consider a network $\mathcal{G}$ with N nodes and F flows. For the network, without loss of generality, we assume the flow rate $x_i \in \mathbf{x}_g$ of each flow is a random variable with mean μ and variance σ^2. Then, we have the following results on the inference error for each template.

Theorem 1: The inference errors (in terms of the genie bound in Defition 2) of templates $T_1 - T_3$ satisfy

1) for the template T_1: k-random-relay,

$$\Theta\left(\frac{F^2\mu^2}{\sqrt{N}/(k+1)+F}\right) \leq I_{T_1} \leq \Theta\left(2F^2(\mu^2+\sigma^2)\right); \tag{9}$$

2) for the template T_2: k-random-neighbor,

$$\Theta\left(\frac{F^2\mu^2}{N/\varphi(N,k)+F}\right) \leq I_{T_2} \leq \Theta\left(2F^2(\mu^2+\sigma^2)\right); \tag{10}$$

3) for the template T_3: k-random-path,

$$\Theta\left(\frac{F^2\mu^2(k-1)^2}{k^2(\sqrt{N}+F)}\right) \leq I_{T_3} \leq \Theta\left(\frac{2F^2(\mu^2+\sigma^2)}{k/(k-1)}\right), \tag{11}$$

where $\varphi(N,k) = \max\{N^{\log(k-1)}\log(k-1)\ln(N), \sqrt{N}\}$

Remark 2: Theorem 1 shows the impact regions of three routing templates. We note that only the upper bound of k-random-path is positively related to k, indicating that when k is small, k-random-relay and k-random-neighbor templates are possible to induce a higher inference error than k-shortest-path. In addition, we find that the bound is an approximately linear (quadratic) function of the number of flows F in the network. This interestingly reveals that increasing the number of flows in the network (i.e., making the communication scenario more complex) in fact incurs more errors than trying to actively creating randomness in routing behavior. A reasonably small k should be chosen in practice due to the sub-linear relationship between the error and k.

2) Delay: We have the following theorem to analyze the costs of delay for three randomized routing templates.

Theorem 2: The delays of templates $T_1 - T_3$ satisfy

1) for the template T_1: k-random-relay,

$$h_{T_1} = \Theta((k+1)\sqrt{N}); \tag{12}$$

2) for the template T_2: k-random-neighbor,

$$\Theta(\varphi(N,k)) \leq h_{T_2} \leq \Theta(N\ln(N)); \tag{13}$$

3) for the template T_3: k-random-path,

$$h_{T_3} = \Theta(\sqrt{N}), \tag{14}$$

where $\varphi(N,k) = \max\{N^{\log(k-1)}\log(k-1)\ln(N), \sqrt{N}\}$.

Remark 3: Theorem 2 shows the delay impacts of the three randomized routing templates. It is noted that the delay of k-random-relay increases linearly with k. When k is small, the lower bound of k-random-neighbor is the same as k-random-path (i.e., $\sqrt{N}$); thus they are generally better than k-random-relay in terms of the delay cost. When k is larger, the delay cost of k-random-neighbor becomes substantially large. This is because when k increases to the average degree of the network, all neighbors can be selected randomly, leading to a random walk behavior of packet forwarding with the $N\log N$ delay cost.

Remark 4: Theorems 1 and 2 show that there is no uniformly best template among the templates in terms of both security and cost. However we notice that k-random-neighbor with a large k is not a good choice for a practical randomized routing design in that (i) it has the highest delay cost compared with others, (ii) the inference error is still on the same order with others; and (iii) for a packet with a limited lifespan (e.g., it can be controlled by Time-to-Live parameter), it cannot guarantee to deliver packets to the destination.

Our results also indicate that k-random-path achieves the inference error with the same order of the other two templates while maintaining the lowest delay cost. But it requires knowing the global path information of the network. For k-random-relay, both the inference error and the delay is larger than k-random-path by a constant order of magnitude.

B. Proofs of Results

Now we prove the theoretical results. With Model 1, we first provide a lemma regarding the generic result of the inference error which depends on the delay overhead.

Lemma 1: For a routing template T satisfying Model 1, in which the average number of columns of its template matrix is denoted by g_T and the delay is denoted by h_T, the inference error I_T induced by T satisfies

$$\Theta\left(\frac{F^2\mu^2(g_T-1)^2}{g_T^2(N/h_T+F)}\right) \leq I_T \leq \Theta\left(\frac{2F^2(g_T-1)}{g_T/(\mu^2+\sigma^2)}\right). \tag{15}$$

Proof: See Appendix. □

Remark 5: Lemma 1 provides a generic impact region of the inference error of any routing template T. We notice that the inference error $I_T = 0$ when $g_T = 1$, meaning that the deterministic routing protocol has no inference error.

Next, we prove Theorem 2 to obtain the delay costs for three templates, then prove Theorem 1 which requires some results in the proof of Theorem 2.

Proof of Theorem 2: We first prove 3): h_{T_3} and 1): h_{T_1}, then we prove 2): h_{T_2}.

3) In the network $\mathcal{G}$, we assume the distance of flow f_i is d_i (the number of hops on the shortest path). For k-random-path template, the average delay for flow f_i is given by $h_{f_i} = d_i + \frac{1}{k}\sum_{j=1}^{k} c_{ij}$, where c_{ij} is a positive constant denoting the

extra number of hops for jth path compared with the shortest one for flow f_i. Then the average is given by

$$h_{T_3} = \frac{1}{F}\sum_{i=1}^{F} d_i + \frac{1}{Fk}\sum_{i=1}^{F}\sum_{j=1}^{k} c_{ij}. \tag{16}$$

According to Lemma 6, we have $\frac{1}{F}\sum_{i=1}^{F} d_i = \Theta(\sqrt{N})$. In addition, it is easy to know $\frac{1}{Fk}\sum_{i=1}^{F}\sum_{j=1}^{k} c_{ij} = \Theta(1)$, then we can obtain $h_{T_3} = \Theta(\sqrt{N})$.

1) For k-random-relay, we randomly select k relays to forward each packet and each path between two consecutive relays adopts the shortest path. Then the average delay can be directly written as $h_{T_3} = \Theta((k+1)\sqrt{N})$.

2) The routing path of k-random-neighbor is discovered hop by hop, which is similar with the random walk (or Markov chain) model [32]–[35]. In the following, we first briefly introduce the difference between the random walk model and k-random-neighbor, then we state how to use the random walk to model the k-random-neighbor template.

In a network $\mathcal{G}$, for an arbitrary node pair $v_i, v_j \in \mathcal{V}$, denote $h_{i,j}$ as the delay for the flow between v_i and v_j. Denote by $\mathcal{N}_i$ the node set containing all neighbors of node v_i. Then the delay cost of the flow between v_i and v_j satisfies the following recursive function

$$h_{i,j} = \begin{cases} 1 + \sum_{w\in\mathcal{N}_i} p_{iw} h_{w,j}, & \text{if } i \neq j \\ 0, & \text{otherwise,} \end{cases} \tag{17}$$

where p_{iw} is the transfer probability that the link between v_i and v_w is selected. In the random walk model, every neighbor in $\mathcal{N}_i$ has the same probability to be selected, i.e,

$$p_{iw} = \frac{1}{|\mathcal{N}_i|}, \text{ for } \forall v_w \in \mathcal{N}_i. \tag{18}$$

For k-random-neighbor with source and destination nodes v_i and v_j, we define a new neighbor set $\mathcal{K}_i \subseteq \mathcal{N}_i$ for node v_i containing k neighbors which have the shortest distances to the destination v_j. Then, the transfer probability p_{iw} can be given by

$$p_{iw} = \frac{1}{k}, \text{ for } \forall v_w \in \mathcal{K}_i. \tag{19}$$

Intuitively, if we can remove all neighbors that belong to $\mathcal{N}_i$ but not to $\mathcal{K}_i$, then $\mathcal{K}_i = \mathcal{N}_i$, and k-random-neighbor becomes the same as the random walk model. To do so, for the original network $\mathcal{G}$, given a source-destination pair v_i and v_j, we generate a new network $\mathcal{G}_{ij}^k = \{\mathcal{V}_{ij}^k, \mathcal{L}_{ij}^k\}$, where $\mathcal{V}_{ij}^k$ is obtained by removing nodes that the source node v_i will never go through under k-random-neighbor for the node pair v_i and v_j. Denote by $\mathcal{L}_{ij}^k$ the corresponding remaining link set for the node set $\mathcal{V}_{ij}^k$. Then, k-random-neighbor can be modeled as a random walk for the network $\mathcal{G}_{ij}^k$.

Considering the new generated network $\mathcal{G}_{ij}^k$, according to (5) and (6) from [36], the expected delay (17) over all node pairs is given by

$$\mathbb{E}(h_{i,j}) = \Theta\left(\mathbb{E}\left(N_{ij}^k \ln\left(\sqrt{N_{ij}^k}\right)\right)\right), \tag{20}$$

where $N_{ij}^k = |\mathcal{V}_{ij}^k|$. For the upper bound, we have $\mathbb{E}(N_{ij}^k) \leq N$. For the lower bound, k-random-path has the smallest delay overhead because it is obtained by averaging k shortest paths, then according to (12), we have that $\mathbb{E}(h_{i,j}) \geq \Theta(\sqrt{N})$. Furthermore, according to Lemma 9, we have

$$\mathbb{E}(h_{i,j}) \geq \begin{cases} \Theta\left(N^{\log_{\delta-1}^{k-1}} \ln\left(N^{\log_{\delta-1}^{k-1}}\right)/2\right), & \text{if } k > 2 \\ \Theta\left(\log(N) \ln\left(\sqrt{\log(N)}\right)\right), & \text{if } k = 2. \end{cases} \tag{21}$$

Combining them together, we obtain

$$\mathbb{E}(h_{i,j}) \geq \max\{N^{\log(k-1)} \log(k-1)\ln(N), \sqrt{N}\}, \tag{22}$$

which completes the proof. □

Proof of Theorem 1: From the generic result (15) in Lemma 1, h_T and $(g_T - 1)/g_T$ depend on routing templates. Note that h_T is available in Theorem 2. Therefore, in the following, we derive $(g_T - 1)/g_T$ under different routing templates.

For k-shortest-relay, we randomly select k relays from the remaining $N-2$ nodes (excluding source and destination nodes). When the node density λ is sufficiently large such that any node pair is connected, we have $g_{T_1} = \Theta(N^k)$, yielding $(g_{T_1} - 1)/g_{T_1} = \Theta(1)$.

For k-shortest-neighbor, at each hop, one neighbor is selected uniformly at random from k possible neighbors. Assume the average distance for a node pair is d, which can be expressed as a function $d = f(N)$. Then, the total number of paths g_{T_2} can be approximated by $k^{f(N)}$. Since $f(N)$ is an increasing function of N, g_{T_2} increases exponentially in N. Therefore, we have $(g_{T_2} - 1)/g_{T_2} = \Theta(1)$.

For k-shortest-path, it is obvious that $g_{T_3} = \Theta(k)$. Then, inserting it into (15) completes the proof. □

V. Experimental Evaluation

In this section, we use simulations to show the inference errors and delay performance under routing templates.

A. Experimental Setups

1) Network Configuration: In experiments, we use RGG to simulate wireless network topologies, where $N \in [20, 100]$ nodes are randomly placed on region $[0, \sqrt{N/\lambda}]$. The node density and transmission range are $\lambda = 5$ and $r = 2$, respectively. In this network, we randomly select $F = \sqrt{N}$ node pairs to have real flow, in which the flow rate of each node pair x_i subjects to the normal distribution with mean $\mu = 10$ and variance $\sigma^2 = 2$.

2) Performance Metrics: Under the worst case that the adversary's inference is unknown, we use the genie bound to measure the inference errors of different routing templates. The method to derive the genie bound is described in Definition 2. The delay is measured by averaging the number of hops for all flows in the network.

3) Routing Template Scenarios: The randomness of each routing template depends on the variable k. Therefore, in our experiments, with respect to k, we consider two different scenarios: (i) fixed k and (ii) dynamic k.

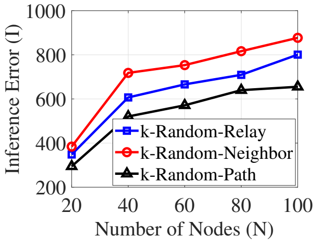


Fig. 5. Inference error for three routing templates when $k = 3$.

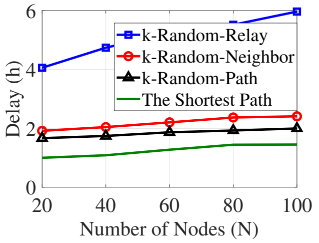


Fig. 6. Delay overhead for three routing templates when $k = 3$.

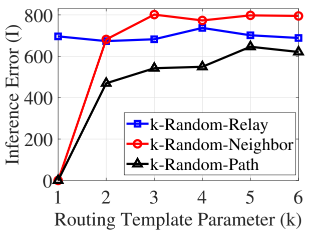


Fig. 7. Inference error for three routing templates when $N = 50$.

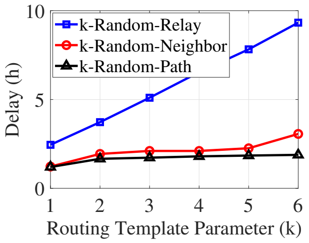


Fig. 8. Delay overhead for three routing templates when $N = 50$.

B. Fixed k Scenario

We first consider the scenario that three templates adopt the fixed $k = 3$. Specifically, For each packet, template T_1: k-random-relay randomly selects 3 relays to forward the packet. For template T_2: k-random-neighbor, each packet is forwarded to one of the 3 neighbors which have the shortest distance to the destination. Template T_3: k-shortest-path directly selects one path from 3 shortest paths for each flow. We apply $F = \sqrt{N}$ into Theorem 1, and obtain that under any routing template, the inference error satisfies $\Theta(\sqrt{N}) \leq I \leq \Theta(N)$.

Fig. 5 depicts the inference error under three randomized routing templates as we change the number of nodes N from 20 to 100. First of all, we see that for every routing template, the inference error increases when the number of nodes increases, which verifies Theorem 1. For instance, the interference error is 718.1 for the k-random-neighbor template when there are 40 nodes. Since each flow on average has the rate 10, we know the attacker has the wrong estimation on at least $718.1/10^2 \approx 7.18$ flows, and this number increases to 8.77 flows when the number of nodes becomes to 100. In addition, we also notice that the difference between templates does not change as we increase N, implying that the difference of induced errors among different templates is on a constant order.

Fig. 6 shows the delay overhead of different routing templates. From Fig. 6, we observe that the slope of the curve of the k-random-relay template is around 4 times that of the shortest path protocol (e.g., when $N = 100$, the delay of k-random-relay and the shortest path is 5.97 and 1.45 respectively). Note that, under the scenario $k = 3$, $N^{\log k} \ln N < (k+1)\sqrt{N}$, therefore, the delay performance of k-random-neighbor is less than k-random-relay.

C. Dynamic k Scenario

For the dynamic k scenario, we fix the number of nodes $N = 50$. The total number of paths increases exponentially for k-random-neighbor as k increases. Fig. 7 depicts the relationship between the inference error and k. We notice that the inference error increases as k increases for all templates. However, as k keeps increasing, the error starts to increase slightly and remains approximate the same value, which verifies Theorem 1 that reveals the inference error increases sub-linearly in k and a large k will not increase the error significantly.

Fig. 8 shows that delay cost of three routing templates with different k. From Fig. 8, we observe that the delay increases linearly for k-random-relay because increasing k results in adding extra paths and linear increase of the delay according to Theorem 2. It is also observed that k-random-neighbor and k-random-path have similar delays, because their delays are on the same order $\Theta(\sqrt{N})$ when $\log k$ is small.

VI. Related Work

A. Random Routing

Due to the dynamics in the wireless environment, network nodes are usually subject to sleep modes [32], channel fluctuation [36], mobility [37], [38], interference from neighbors [39]. Therefore, to account for such stochastic and asymmetric natures, random routing strategies and their performance have been widely studied in wireless networks [29], [32]–[36]. For example, [32]–[36] leveraged the random walk to model randomized routing strategies and evaluate the delay and packet delivery ratio. Recently, the work in [21] presents the initial results that being randomized can help security and cause substantial errors in malicious network inference. However, there is still a lack of study regarding how exactly a randomized protocol should be designed to protect network flow infomration from being disclosed. In this paper, we categorize randomized routing into three templates k-random-relay, k-random-neighbor and k-random-path, and carefully analyze the inference error and delay cost of each template.

B. Security of Network Inference

Network flow based attack and defense strategies have been explored in a line of works [5], [7], [10]–[12], [25], [26], [40]–[43]. However, in the wireless network, the PHY or MAC layer activities are public, therefore the flow information is still achievable through the network inference, which is a family of network monitoring techniques that build the network characteristics from indirect measurements [13]–[17]. Existing studies on the network inference mainly focused on optimizing inference methods to obtain more information from given measurements. For example, authors in [15], [18] proposed new ways to detect anomaly links. In [42], authors proposed a faster and practical interference method by combining the tomography and gravity; and a more accurate interference method is investigated in [43] when the measurement frequency is changing. The improved inference methods in fact open up an opportunity for the adversary to obtain the flow information. For example, the work in [6] proposed an attack strategy to retrieve the traffic pattern by using interatrial based traffic analysis. Therefore, in this paper, we investigate the fundamental reason of such vulnerability and find the randomized routing protocol can lead to more inference errors, so that protect the real flow information against leakage.

VII. Conclusion

In this paper, we revisited the network inference and found that the inference accuracy depends on routing protocols. We categorized randomized routing protocols into three templates, k-random-relay, k-random-neighbor and k-random-path, and theoretically analyzed the inference errors and incurred delay costs of them. We conducted simulations to confirm that randomized routing templates can inflict different inference errors and delays, yielding different capabilities to prevent the flow information leakage.

Acknowledgement: The work at University of South Florida was supported in part by NSF 1717969.

Appendix A

In this section, we first prove Lemma 1, then we introduce and prove other necessary lemmas.

A. Proof of Lemma 1

Proof: From Model 1, considering the worst case that both $\mathbf{M}$ and $\mathbf{T}$ are available to the adversary, for real flows, we have the linear system

$$\mathbf{y} = \mathbf{RC}\mathbf{x}_g, \tag{23}$$

where $\mathbf{R} = \mathbf{MT}$ is the routing matrix under the template represented by the matrix $\mathbf{T}$. As aforementioned, the selection matrix $\mathbf{C}$ is unknown by the adversary, then we let $\mathbf{D} = \{\mathbf{d}_i\} \in \mathbb{R}^{\sum g_i \times F}$ be the selection matrix of the adversary, that randomly selects a path for any flow. Considering the worst case that the ground-truth matrix $\mathbf{C}$ and $\mathbf{D}$ have the same dimension, then from the adversary perspective, the linear system (23) can be written as

$$\mathbf{y} = \mathbf{RD}\mathbf{x}_g. \tag{24}$$

Taking the least square estimation of $\mathbf{x}_g$, we obtain $\hat{\mathbf{x}}_g = [(\mathbf{RD})^T\mathbf{RD}]^{-1}(\mathbf{RD})^T\mathbf{y}$, then putting it into the genie bound metric (7), we have that

$$\begin{aligned} G(\mathbf{x}_g) &= \mathbb{E}\left(\|\hat{\mathbf{x}}_g - \mathbf{x}_g\|_2^2\right) \\ &= \mathbb{E}\left(\|[(\mathbf{RD})^T\mathbf{RD}]^{-1}(\mathbf{RD})^T(\mathbf{RC}\mathbf{x}_g) - \mathbf{x}_g\|_2^2\right) \\ &= \mathbb{E}\left(\|[(\mathbf{RD})^T\mathbf{RD}]^{-1}(\mathbf{RD})^T(\mathbf{RC}) - I]\mathbf{x}_g\|_2^2\right) \\ &= \mathbb{E}\left(\|\mathbf{U}[\mathbf{RC} - \mathbf{RD}]\mathbf{x}_g\|_2^2\right), \end{aligned} \tag{25}$$

where $\mathbf{U} = [(\mathbf{RD})^T\mathbf{RD}]^{-1}(\mathbf{RD})^T$. Then by leveraging Lemma 3, the following inequality holds with high probability.

$$\begin{aligned} &\lambda_{\min}(\mathbf{U}^T\mathbf{U})\|\Delta\mathbf{x}_g\|_2^2 \\ \le&\|\mathbf{U}[\mathbf{RC} - \mathbf{RD}]\mathbf{x}_g\|_2^2 \le \lambda_{\max}(\mathbf{U}^T\mathbf{U})\|\Delta\mathbf{x}_g\|_2^2. \end{aligned} \tag{26}$$

From Lemma 4, $\lambda_{\min}(\mathbf{U}^T\mathbf{U})$ and $\lambda_{\max}(\mathbf{U}^T\mathbf{U})$ can be replaced by $\lambda_{\max}^{-1}(\mathbf{RD}(\mathbf{RD})^T)$ and $\lambda_{\min}^{-1}(\mathbf{RD}(\mathbf{RD})^T)$ respectively, then (25) can be expressed as the following inequality with high probability

$$\begin{aligned} &\mathbb{E}\left(\frac{\|\Delta\mathbf{x}_g\|_2^2}{\lambda_{\max}(\mathbf{RD}(\mathbf{RD})^T)}\right) \\ \le&G(\mathbf{x}_g) \le \mathbb{E}\left(\frac{\|\Delta\mathbf{x}_g\|_2^2}{\lambda_{\min}(\mathbf{RD}(\mathbf{RD})^T)}\right), \end{aligned} \tag{27}$$

where $\Delta = \mathbf{RC} - \mathbf{RD}$. Then leveraging Lemma 5, we can derive the following asymptotically solution

$$\frac{\mathbb{E}\|\Delta\mathbf{x}_g\|_2^2}{\Theta\left(h(N) + \frac{Fh(N)^2}{N}\right)} \le I_T \le \frac{\mathbb{E}\|\Delta\mathbf{x}_g\|_2^2}{\Theta(h(N))}. \tag{28}$$

Now we derive the numerator. Denote entries in Δ as $\{\delta_{ij}\}_{i\in[1,L],j\in[1,F]}$, where each entry $\delta_{ij} \in \{1, -1, 0\}$. Because the adversary randomly select one path for each flow, we have that

$$\begin{aligned} \Pr\{\delta_{ij} = 1\} =& \Pr\{\delta_{ij} = 1|\mathbf{c}_j \neq \mathbf{d}_j\}\Pr\{\mathbf{c}_j \neq \mathbf{d}_j\} \\ =& \frac{g_j - 1}{g_j}\Pr\{\delta_{ij} = 1|\mathbf{c}_j \neq \mathbf{d}_j\} \\ =& \frac{g_j - 1}{g_j}\Theta\left(\frac{h(N)}{N}\right)\left(1 - \Theta\left(\frac{h(N)}{N}\right)\right). \end{aligned} \tag{29}$$

Furthermore, it is easy to know $\Pr\{\delta_{ij} = -1\} = \Pr\{\delta_{ij} = 1\}$. Since all nodes are uniformly distributed in a region Ω. Then for routing template T, $\mathbb{E}(g_i) = g_T$ for $1 \le i \le F$, and we have the following two equations:

$$\begin{aligned} \mathbb{E}\{\delta_{ij}\} &= \Pr\{\delta_{ij} = 1\} - \Pr\{\delta_{ij} = -1\} \\ &= \frac{g_T - 1}{g_T}\Theta\left(\frac{h(N)}{N}\right), \end{aligned} \tag{30}$$

and

$$\begin{aligned} \mathbb{E}\{\delta_{ij}^2\} &= \Pr\{\delta_{ij} = 1\} + \Pr\{\delta_{ij} = -1\} \\ &= \frac{2(g_T - 1)}{g_T}\Theta\left(\frac{h(N)}{N}\right). \end{aligned} \tag{31}$$

Then, we have the lower bound as

$$\begin{aligned} \mathbb{E}\|\Delta\mathbf{x}_g\|_2^2 &= \mathbb{E}\left(\sum_{i=1}^{L}\left(\sum_{j=1}^{F}\delta_{ij}x_j\right)^2\right) \\ &= \Theta(N)\mathbb{E}\left(\left(\sum_{j=1}^{F}\delta_{ij}x_j\right)^2\right) \\ &\ge \Theta(N)\left(\mathbb{E}\left(\sum_{j=1}^{F}\delta_{ij}x_j\right)\right)^2 \\ &= \Theta\left(\frac{[F\mu(g_T - 1)h(N)]^2}{g_T^2 N}\right). \end{aligned} \tag{32}$$

On the other hand, by using Cauchy-Schwarz inequality, the upper bound is

$$\begin{aligned}\mathbb{E}\|\Delta \mathbf{x}_g\|_2^2 &= \Theta(N)\mathbb{E}\left(\left(\sum_{j=1}^{F} \delta_{ij} x_j\right)^2\right)\\ &\le \Theta(N)\mathbb{E}\left(\sum_{j=1}^{F} \delta_{ij}^2\right)\mathbb{E}\left(\sum_{j=1}^{F} x_j^2\right)\\ &= \Theta\left(\frac{2F^2(g_T-1)(\mu^2+\sigma^2)h(N)}{g_T}\right).\end{aligned} \tag{33}$$

Replacing (32), (33) into (28), we complete the proof. □

B. Proof of Necessary Lemmas

Lemma 2: The probability that each element in the matrix $\mathbf{R}$ is

$$\Pr\{r_{ij}=1\} = \Theta\left(\frac{h(N)}{N}\right). \tag{34}$$

Proof: From Model 1, we have that

$$\Pr\{r_{ij}=1\} = \Pr\{\text{path } p_j \text{ traverses link } l_i\}. \tag{35}$$

We assume there are u links in the network and path p_j contains u_j links, then (35) can be rewritten as

$$\Pr\{\text{flow } f_j \text{ traverses link } l_i | u, u_j\} = u_j/u. \tag{36}$$

Take the expectation to u and u_j, we have that

$$\Pr\{r_{ij}=1\} = \mathbb{E}_{u_j}\left(\mathbb{E}_u\left(\frac{u_j}{u}|u_j\right)\right). \tag{37}$$

From Taylor series at $\mathbb{E}(u|u_j)$, we have

$$\mathbb{E}_u\left(\frac{u_j}{u}|u_j\right) = \frac{u_j}{\mathbb{E}_u(u|u_j)} + f\left(O\left(\frac{\text{Var}_u(u|u_j)}{\mathbb{E}^3(u|u_j)}\right)\right). \tag{38}$$

By leveraging Lemma 7, (37) can be derived as

$$\Pr\{r_{ij}=1\} = \mathbb{E}\left(\frac{u_j}{N(\lambda\pi r^2-1)/2}\right) = \frac{\mathbb{E}(u_j)}{\Theta(N)}. \tag{39}$$

According to Definition 1, $\mathbb{E}(u_j) = h(N)$, then we complete the proof. □

Lemma 3: For a matrix $\mathbf{A}$, it satisfies

$$\lambda_{\min}(\mathbf{A})\|\alpha\|_2^2 \le \|\mathbf{A}\alpha\|_2^2 \le \lambda_{\max}(\mathbf{A})\|\alpha\|_2^2, \tag{40}$$

where $\lambda_{\min}(\mathbf{A})$ and $\lambda_{\max}(\mathbf{A})$ is the minimum and maximum eigenvalues of matrix $\mathbf{A}$.

Proof: Clearly, we know

$$\|\mathbf{A}\alpha\|_2^2 = \|\alpha^T\mathbf{A}^T\mathbf{A}\alpha\| = \frac{\|\alpha^T\mathbf{A}^T\mathbf{A}\alpha\|}{\alpha^T\alpha}\|\alpha\|_2^2. \tag{41}$$

According to [44], $\frac{\|\alpha^T\mathbf{A}^T\mathbf{A}\alpha\|}{\alpha^T\alpha}$ has the minimum value $\lambda_{\min}(\mathbf{A}^T\mathbf{A})$ and the maximum value $\lambda_{\max}(\mathbf{A}^T\mathbf{A})$, then we can complete the proof. □

Lemma 4: For a matrix $\mathbf{A} \in \mathbb{R}^{m\times n}$ where $m > n$, let $\mathbf{G} = (\mathbf{A}^T\mathbf{A})^{-1}\mathbf{A}^T$, then $\lambda_{\max}(\mathbf{G}^T\mathbf{G}) = \lambda_{\min}^{-1}(\mathbf{A}^T\mathbf{A})$ and $\lambda_{\min}(\mathbf{G}^T\mathbf{G}) = \lambda_{\max}^{-1}(\mathbf{A}^T\mathbf{A})$.

Proof: Based on the singular value decomposition, for the matrix $\mathbf{A} \in \mathbb{R}^{m\times n}$, there exists a couple of unitary matrices $\mathbf{U} \in \mathbb{R}^{m\times m}$ and $\mathbf{V} \in \mathbb{R}^{n,n}$, such that $\mathbf{A} = \mathbf{U}\Lambda\mathbf{V}^T$ where $\Lambda = \text{diag}(s_1, \cdots, s_n) \in \mathbb{R}^{m\times n}$ is a rectangular diagonal matrix. The numbers $s_i = \lambda_i(\sqrt{\mathbf{A}^T\mathbf{A}})$ for $i = 1, \cdots, n$ are singular values of $\mathbf{A}$. If one sees the diagonal matrix $D := \text{diag}(s_1^2, \cdots, s_n^2) \in \mathbb{R}^{n\times n}$, we have

$$\mathbf{A}^T\mathbf{A} = \mathbf{V}D\mathbf{V}^T. \tag{42}$$

Then $(\mathbf{A}^T\mathbf{A})^{-1} = \mathbf{V}D^{-1}\mathbf{V}^T$, and $\mathbf{G}$ can be derived as

$$\mathbf{G} = (\mathbf{A}^T\mathbf{A})^{-1}\mathbf{A}^T = \mathbf{V}D^{-1}\mathbf{V}^T\mathbf{V}\Lambda\mathbf{U}^T = \mathbf{V}\Lambda^{-1}\mathbf{U}^T, \tag{43}$$

where $\Lambda^{-1} = \text{diag}(s_1^{-1}, \cdots, s_n^{-1}) \in \mathbb{R}^{n\times m}$. Similarly,

$$\mathbf{G}^T\mathbf{G} = \mathbf{U}(\Lambda^{-1})^2\mathbf{U}^T = \mathbf{U}D^{-1}\mathbf{U}^T. \tag{44}$$

Comparing (42) and (44) we know that $\lambda(\mathbf{G}^T\mathbf{G}) = \lambda^{-1}(\mathbf{A}^T\mathbf{A})$, which completes the proof. □

Lemma 5: For a random binary matrix $\mathbf{A}$ with size $L \times F$, where the expectation of each entry of $\mathbf{A}$ is $\mathbb{E}(a_{ij}) = \Theta(h(N)/N)$ for $h(N) = O(N)$ and $L = \Theta(N)$. Then if $F \to \infty$ with $\lim_{L\to\infty} F/L < \infty$, then the following two statements are satisfied with high probability,

1) for the minimum eigenvalue, $\lambda_{\min}(\mathbf{A}^T\mathbf{A}) = \Theta(h(N))$
2) for the maximum eigenvalue,

$$\lambda_{\max}(\mathbf{A}^T\mathbf{A}) \le \Theta\left(h(N) + \frac{Fh^2(N)}{N}\right)$$

Proof: For the statement (1), from Lemma 8, we have a function f such that the $\text{Var}(f\mathbf{A}) = 1$. Then statement (1) can be rewritten as

$$\lambda_{\min}(\mathbf{A}^T\mathbf{A}) = \frac{L}{f^2}\lambda_{\min}(L^{-1}(c\mathbf{A})^T(c\mathbf{A})). \tag{45}$$

From [44], we have $\lambda_{\min}(L^{-1}(c\mathbf{A})^T(c\mathbf{A})) = \Theta(1)$ with high probability. Then (45) can be further derived as

$$\lambda_{\min}(\mathbf{A}^T\mathbf{A}) = \frac{L}{f^2}\Theta(1) = \frac{\Theta(N)}{\Theta(h(N)/N)} = \Theta(h(N)). \tag{46}$$

For the statement (2), we define two matrix $\mathbf{U}$ and $\mathbf{V}$ where $\mathbf{V}$ is an all-one matrix and each entry in $u_i \in \mathbf{U}$ satisfies $\mathbb{E}(u_i) = 0$. Then we have $\mathbf{A} = \mathbf{U} + \mathbf{V}h(N)/N$. Applying to $\mathbf{A}^T\mathbf{A}$, we have that

$$\begin{aligned}\lambda_{\max}(\mathbf{A}^T\mathbf{A}) &= \lambda_{\max}((\mathbf{U}^T + \mathbf{V}^T h(N)/N)(\mathbf{U} + \mathbf{V}h(N)/N))\\ &\le \lambda_{\max}(\mathbf{U}^T\mathbf{U}) + 2h(N)/N\lambda_{\max}(\mathbf{U}^T\mathbf{V})\\ &\quad + (h(N)/N)^2\lambda_{\max}(\mathbf{V}^T\mathbf{V}).\end{aligned} \tag{47}$$

For the second term in (47), since $\mathbf{V}$ is an all-one matrix with rank 1, then we have that

$$\lambda_{\max}(\mathbf{U}^T\mathbf{V}) = \text{tr}\{\mathbf{U}^T\mathbf{V}\} = \sum_{u_{ij}\in\mathbf{U}} u_{ij}. \tag{48}$$

According to the large number law, $\lambda_{\max}(\mathbf{U}^T\mathbf{V}) = o(NF)^{1/p}$ for $1 < p < 2$. Then similarly,

$$\lambda_{\max}(\mathbf{V}^T\mathbf{V}) = \sum_{v_{ij}\in\mathbf{V}} 1 = \Theta(FN) \tag{49}$$

From [44], the first term satisfy $\lambda_{\max}(\mathbf{U}^T\mathbf{U}) = \Theta(h(N))$. Replacing $\lambda_{\max}(\mathbf{U}^T\mathbf{V})$, $\lambda_{\max}(\mathbf{V}^T\mathbf{V})$, $\lambda_{\max}(\mathbf{U}^T\mathbf{U})$ into (47), we can complete the proof. □

Lemma 6: For network $\mathcal{G}$ with N nodes, the average delay over all flows by leveraging the shortest path protocol is $d = \Theta(\sqrt{N})$.

Proof: The delay is positively related to the distance between the source and destination nodes. Let e_i be the Euclidean distance for flow f_i. Since each hop covers a distance of $\Theta(r)$, the delay of flow by using the shortest path is f_i is $\Theta(e_i/r)$. The average delay over all flows is $\Theta(\frac{1}{N}\sum_{i=1}^{N} e_i/r)$. Since all nodes are randomly distributed in a region $\Omega = [0, \sqrt{N/\lambda}]^2$, for a large N, we have $\frac{1}{N}\sum_{i=1}^{N} e_i = \Theta\left(\sqrt{N/\lambda}\right)$. Therefore the average delay can be derived as $d = \Theta\left(\frac{\sqrt{N/\lambda}}{r}\right) = \Theta(\sqrt{N})$, which completes the proof. □

Lemma 7: In a RGG with N nodes, L links, and density λ and transmission range r, then L is on the order of N with high probability, i.e., $\Pr\{L = \Theta(N)\} = 1 - \Theta(e^{-\Theta(1)N})$.

Proof: In RGG, the degree of an arbitrary node equals to the number of nodes dropping into the transmission range of this node, then the degree of an arbitrary subjects to Poisson distribution with parameter $\lambda\pi r^2 - 1$, and thus the distribution of the total number of link is also Poisson distribution with parameter $N(\lambda\pi r^2 - 1)/2$. Then based on the Chernoff bound, for a small constant $c_1 < (\lambda - 1)\pi r^2$ and a large constant $c_2 > (\lambda - 1)\pi r^2$, we have the following upper and lower bounds

$$\begin{aligned}\Pr\{L \geq c_1 N\} &\leq 1 - \frac{e^{\frac{-N(\lambda\pi r^2-1)}{2}}\left(\frac{eN\lambda(\pi r^2-1)}{2}\right)^{c_1N}}{(c_1N)^{c_1N}} \\ &= 1 - e^{\frac{-N(\lambda\pi r^2-1)}{2}} e^{c_1N\log\left(\frac{e\lambda(\pi r^2-1)}{2c_1}\right)} \\ &= 1 - \Theta(e^{-\Theta(1)N})\end{aligned} \tag{50}$$

and

$$\begin{aligned}\Pr\{L \leq c_2 N\} &\geq 1 - \frac{e^{\frac{-N(\lambda\pi r^2-1)}{2}}\left(\frac{eN\lambda(\pi r^2-1)}{2}\right)^{c_2N}}{(c_2N)^{c_2N}} \\ &= 1 - \Theta(e^{-\Theta(1)N})\end{aligned} \tag{51}$$

where $\xi = \Theta(e^{-\Theta(1)N})$. Then we complete the proof. □

Lemma 8: For the routing matrix $\mathbf{R}$ in Lemma 2, there exists a function $f = \sqrt{N/h(N)}$ such that each entry in $\mathbf{R}$ has finite mean and invariance 1.

Proof: Denote each entry in $f\mathbf{R}$ as e_{ij}, then the mean of e_{ij} can be derived as

$$\mathbb{E}(e_{ij}) = \mathbb{E}(fr_{ij}) = f\Pr\{r_{ij} = 1\} = f\Theta\left(\sqrt{\frac{h(N)}{N}}\right) = \Theta(1), \tag{52}$$

and the variance is

$$\begin{aligned}\text{Var}(e_{ij}) &= \mathbb{E}(e_{ij}^2) - \mathbb{E}^2(e_{ij}) = f^2\mathbb{E}(r_{ij}^2) - \Theta(1) \\ &= f^2\Theta\left(\frac{h(N)}{N}\right) - \Theta(1) = \Theta(1).\end{aligned} \tag{53}$$

Then we can complete the proof. □

Lemma 9: For a network $\mathcal{G}$ with N nodes and its corresponding generated network $\mathcal{G}_{ij}^k$ with N_{ij}^k nodes in the proof of Theorem 2. We have the lower bound $\mathbb{E}(N_{ij}^k) > \Theta(N^{\log_{\delta-1}^{k-1}})$ if $k > 2$ and $\mathbb{E}(N_{ij}^k) > \Theta(\log(N))$ if $k = 2$.

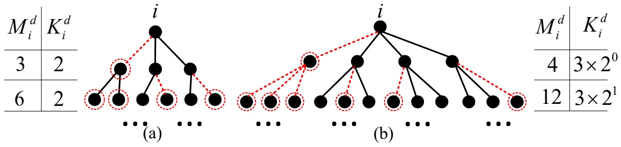


Fig. 9. Example of network topology, where (a) $\delta = 3$, and $k = 2$, and (b) $\delta = 4$, and $k = 3$. Dotted lines and circles denote removed links and nodes.

Proof: For source node v_i, we define node set $\mathcal{M}_i^d$ containing nodes which have distance d with the source node v_i, and $\mathcal{M}_i^0 = \{v_i\}$. Similarly, $\mathcal{K}_i^d$ be the corresponding node set after removing nodes from $\mathcal{M}_i^d$ based on the rule in the proof of Theorem 2. Let $|\mathcal{M}_i^d| = M_i^d$ and $|\mathcal{K}_i^d| = K_i^d$. Since the network is connected, then we have that

$$\sum_{d=0}^{l} M_i^d = N, \quad \sum_{d=0}^{l} K_i^d = N_{ij}^k, \tag{54}$$

where l is the maximum distance.

To derive the low bound of $\mathbb{E}(N_{ij}^k)$, we should remove the most nodes from N nodes. To do so, we consider an extreme case satisfies two requirements:

1) the destination node is located at the maximum distance l away from the source node v_i,
2) the network is a tree structure, i.e., except for the source node, for any node $v_d \in \mathcal{M}_i^d$, it connects with only one node $v_{d-1} \in \mathcal{M}_i^{d-1}$ (as shown in Fig. 9).

Under this case, once a node $v_d \in \mathcal{M}_i^d$ is removed, all nodes that are connected with this node in $\mathcal{M}_i^{d+j}$ for $j \geq 1$ will be removed. Then the expected number of nodes with distance d can be given by

$$\begin{cases}\mathbb{E}(M_i^d) = \delta(\delta-1)^{(d-1)}, & \text{for the network } \mathcal{G} \\ \mathbb{E}(K_i^d) \geq k(k-1)^{(d-1)}, & \text{for the network } \mathcal{G}_{ij}^k.\end{cases} \tag{55}$$

Then we sum all distances together. For networks $\mathcal{G}$ and $\mathcal{G}_{ij}^k$, according to (55), we obtain two equations

$$\sum_{d=1}^{l} \mathbb{E}(M_i^d) = N - 1 = \sum_{d=1}^{l} \delta(\delta-1)^{(d-1)}, \tag{56}$$

and

$$\sum_{d=1}^{l} \mathbb{E}(K_i^d) = \mathbb{E}(N_{ij}^k) - 1 \geq \sum_{d=1}^{l} k(k-1)^{(d-1)}. \tag{57}$$

Combining (56) and (57) together, we can derive the lower bound of $\mathbb{E}(N_{ij}^k)$. Note that according to (57), $k = 2$ is a special scenario, therefore we split our results into two scenarios, i.e., $k = 2$ and $k > 2$. Fig. 9 shows an illustrative example of $k = 2$ and $k = 3$. Then after jointly manipulating (56) and (57), we complete the proof. □

References

[1] M. K. Hanawal, D. N. Nguyen, and M. Krunz, "Jamming attack on in-band full-duplex communications: Detection and countermeasures," in *IEEE INFOCOM*, 2016.

[2] Z. Zhang and A. Mukherjee, "Friendly channel-oblivious jamming with error amplification for wireless networks," in *IEEE INFOCOM*, 2016.

[3] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "Wireless sensor networks: a survey," *Computer networks*, vol. 38, 2002.

[4] C. K. Toh, *Ad hoc mobile wireless networks: protocols and systems*. Pearson Education, 2001.

[5] D. Das, S. Meiser, E. Mohammadi, and A. Kate, "Anonymity trilemma: Strong anonymity, low bandwidth overhead, low latency—choose two," in *IEEE S&P*, 2018.

[6] Y. Liu, R. Zhang, J. Shi, and Y. Zhang, "Traffic inference in anonymous manets," in *IEEE SECON*, 2010.

[7] X. Wang, S. Chen, and S. Jajodia, "Network flow watermarking attack on low-latency anonymous communication systems," in *IEEE S&P*, 2007.

[8] Y. Zhang, W. Liu, W. Lou, and Y. Fang, "Mask: anonymous on-demand routing in mobile ad hoc networks," *IEEE Trans. Wirel. Commun.*, vol. 5, 2006.

[9] Y. Zhang, W. Liu, and W. Lou, "Anonymous communications in mobile ad hoc networks," in *IEEE INFOCOM*, 2005.

[10] P. Syverson, R. Dingledine, and N. Mathewson, "Tor: The secondgeneration onion router," in *USENIX Security*, 2004.

[11] M. K. Reiter and A. D. Rubin, "Crowds: Anonymity for web transactions," *ACM TISSEC*, vol. 1, 1998.

[12] P. Venkitasubramaniam and L. Tong, "A game-theoretic approach to anonymous networking," *IEEE/ACM Trans. Netw.*, vol. 20, 2012.

[13] A. Soule, A. Lakhina, N. Taft, K. Papagiannaki, K. Salamatian, A. Nucci, M. Crovella, and C. Diot, "Traffic matrices: balancing measurements, inference and modeling," in *ACM SIGMETRICS*, 2005.

[14] Z. Zhang, O. Mara, and K. Argyraki, "Network neutrality inference," in *ACM SIGCOMM*, 2014.

[15] T. He, "Distributed link anomaly detection via partial network tomography," 2018.

[16] S. Ahuja, S. Ramasubramanian, and M. Krunz, "SRLG failure localization in all-optical networks using monitoring cycles and paths." in *IEEE INFOCOM*, 2008.

[17] A. Coates, A. O. Hero III, R. Nowak, and B. Yu, "Internet tomography," *IEEE Signal Process. Mag.*, vol. 19, 2002.

[18] A. Lakhina, M. Crovella, and C. Diot, "Diagnosing network-wide traffic anomalies," in *ACM SIGCOMM*, 2004.

[19] M. Mardani and G. B. Giannakis, "Estimating traffic and anomaly maps via network tomography," *IEEE/ACM Trans. Netw.*, vol. 24, 2016.

[20] H. Kasai, W. Kellerer, and M. Kleinsteuber, "Network volume anomaly detection and identification in large-scale networks based on online time-structured traffic tensor tracking," *IEEE Trans. Netw. Service Manag.*, vol. 13, 2016.

[21] Z. Lu and C. Wang, "Network anti-inference: A fundamental perspective on proactive strategies to counter flow inference," in *IEEE INFOCOM*, 2015.

[22] J. W. Guck, A. Van Bemten, M. Reisslein, and W. Kellerer, "Unicast QoS routing algorithms for SDN: A comprehensive survey and performance evaluation," *Commun. Surveys Tuts.*, vol. 20, 2018.

[23] W. Dai and O. Milenkovic, "Subspace pursuit for compressive sensing signal reconstruction," *IEEE Trans. Inf. Theory*, vol. 55, 2009.

[24] S. Boccaletti, V. Latora, Y. Moreno, M. Chavez, and D.-U. Hwang, "Complex networks: Structure and dynamics," *Physics reports*, 2006.

[25] P. Barford and D. Plonka, "Characteristics of network traffic flow anomalies," in *ACM SIGCOMM*, 2001.

[26] A. Houmansadr, N. Kiyavash, and N. Borisov, "RAINBOW: A robust and invisible non-blind watermark for network flows." in *NDSS*, 2009.

[27] "Wireless LAN medium access control (MAC) and physical layer (PHY) specifications," *IEEE Std 802.11*, 2013.

[28] C. E. Perkins and E. M. Royer, "Ad-hoc on-demand distance vector routing," in *IEEE WMCSA*, 1999.

[29] D. C. Dhanapala, A. P. Jayasumana, and Q. Han, "Performance of random routing on grid-based sensor networks," in *IEEE CCNC*, 2009.

[30] B. Karp and H.-T. Kung, "GPSR: Greedy perimeter stateless routing for wireless networks," in *ACM MobiCom*, 2000.

[31] E. J. Candès, J. Romberg, and T. Tao, "Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information," *IEEE Trans. Inf. Theory*, vol. 52, 2006.

[32] C.-K. Chau and P. Basu, "Exact analysis of latency of stateless opportunistic forwarding," in *IEEE INFOCOM*, 2009.

[33] L. Lovász *et al.*, "Random walks on graphs: A survey," *Combinatorics, Paul erdos is eighty*, vol. 2, 1993.

[34] F. Chung and S.-T. Yau, "Discrete green's functions," *Elsevier Journal of Combinatorial Theory, Series A*, vol. 91, 2000.

[35] Y. Li and Z.-L. Zhang, "Random walks on digraphs: A theoretical framework for estimating transmission costs in wireless routing," in *IEEE INFOCOM*, 2010.

[36] I. Mabrouki, X. Lagrange, and G. Froc, "Random walk based routing protocol for wireless sensor networks," in *ICST ValueTools*, 2007.

[37] A. E. Gamal, J. Mammen, B. Prabhakar, and D. Shah, "Throughput-delay trade-off in wireless networks," in *IEEE INFOCOM 2004*, 2004.

[38] M. Grossglauser and D. N. Tse, "Mobility increases the capacity of ad hoc wireless networks," *IEEE/ACM Trans. Netw.*, vol. 10, 2002.

[39] M. J. Neely and E. Modiano, "Capacity and delay tradeoffs for ad hoc mobile networks," *IEEE Trans. Inf. Theory*, vol. 51, 2005.

[40] N. Gelernter and A. Herzberg, "On the limits of provable anonymity," in *ACM WPES*, 2013.

[41] P. Venkitasubramaniam, T. He, and L. Tong, "Anonymous networking amidst eavesdroppers," *IEEE Trans. Inf. Theory*, vol. 54, 2008.

[42] Y. Zhang, M. Roughan, N. Duffield, and A. Greenberg, "Fast accurate computation of large-scale ip traffic matrices from link loads," in *ACM SIGMETRICS*, 2003.

[43] K. Xie, C. Peng, X. Wang, G. Xie, J. Wen, J. Cao, D. Zhang, and Z. Qin, "Accurate recovery of internet traffic data under variable rate measurements," *IEEE/ACM Trans. Netw.*, vol. 26, 2018.

[44] A. M. Sengupta and P. P. Mitra, "Distributions of singular values for some random matrices," *Physical Review E*, vol. 60, 1999.

Shangqing Zhao received his B.S. degree from Fujian Agriculture and Forestry University, Fuzhou, China, in 2010; his M.S. degree from Henan Polytechnic University, Jiaozuo, China, in 2015. He is working toward the Ph. D. degree in the Department of Electrical Engineering, University of South Florida. His research interests include network and mobile system design and security. He is a student member of IEEE and ACM.

Zhuo Lu is an Assistant Professor in the Department of Electrical Engineering, University of South Florida. He currently leads the Communications, Security, and Analytics (CSA) Lab. His research has been supported by NSF, ARO, ONR, DOE and Florida Center for Cybersecurity. Dr. Lu received his B.S. and M.S. degrees from Xidian University, Xi'an China, in 2002 and 2005, respectively, and his Ph.D. degree in computer engineering from North Carolina State University, Raleigh NC, in 2013. Dr. Lu's research has been mainly focused on modeling and analytical perspectives on communication, network, and security. His recent research is equally focused on practical and system perspectives on networking and security. He is a member of IEEE, ACM and USENIX.

Cliff Wang is the division chief of ARO Computing Sciences division. He heads the division staff and manages resources to execute the Army's basic research investment in Computing Sciences with an annual budget of over 20 million dollars. Dr. Cliff Wang graduated from North Carolina State University with a PhD in computer engineering in 1996. He has been carrying out research in the area of computer vision, medical imaging, high speed networks, and most recently information security. He has authored over 50 technical papers and 3 Internet standards RFCs. Dr. Wang also authored/edited for 18 books in the area of information security and hold 4 US patents on information security system development. For the past decades, Dr. Wang managed over $200M research funding which led to significant technology breakthroughs. Dr. Wang also holds adjunct professor appointment at both Department of Computer Science and Department of Electrical and Computer Engineering at North Carolina State University. Dr. Wang is a Fellow of IEEE.