MMP: A Dynamic Routing Protocol Design to Proactively Defend against Wireless Network Inference Attacks

Jinmiao Chen
University of Oklahoma
Norman, OK, USA
jinmiao.chen-1@ou.edu

Zhengping Jay Luo
Rider University
Lawrenceville, NJ, USA
zluo@rider.edu

Yuchen Liu
North Carolina State University
Raleigh, NC, USA
yuchen.liu@ncsu.edu

Shangqing Zhao
University of Oklahoma
Tulsa, OK, USA
shangqing@ou.edu

ABSTRACT

Network inference refers to the process of extracting sensitive information from a network without directly accessing it. This poses a significant threat to network security since it allows attackers to gain insight into sensitive information such as flow information through inference. Possessing flow information about a wireless network can empower attackers to launch more sophisticated and targeted attacks. Network inference relies on consistent traffic patterns or behavior to establish the relationship between the measured link metrics and flow information. Therefore, dynamic routing can help enhance resilience against network inference by proactive introducing variability into network traffic patterns, which can incur a high probability of mismatch between the observed patterns and the actual ones. In this paper, we observe that the inference error is positively related to the mismatch. Therefore, we propose a dynamic routing protocol, called Max-Mismatch-Probability (MMP), which seeks to maximize mismatch probability and increase the inference error. In this paper, we provide the theoretical analysis of our proposed protocol and show that the inference error of MMP is $\Theta(\sqrt{N})$, which is verified in our experimental results.

CCS CONCEPTS

• **Networks** → **Network layer protocols**; • **Security and privacy** → **Mobile and wireless security**.

KEYWORDS

Network inference; dynamic routing; wireless network; proactive defense

ACM Reference Format:
Jinmiao Chen, Zhengping Jay Luo, Yuchen Liu, and Shangqing Zhao. 2023. MMP: A Dynamic Routing Protocol Design to Proactively Defend against Wireless Network Inference Attacks. In *Proceedings of the 10th ACM Workshop on Moving Target Defense (MTD '23), November 26, 2023, Copenhagen, Denmark.* ACM, New York, NY, USA, **??** pages. https://doi.org/10.1145/3605760.3623766

MTD '23, November 26, 2023, Copenhagen, Denmark

ACM ISBN 979-8-4007-0256-3/23/11.
https://doi.org/10.1145/3605760.3623766

1 INTRODUCTION

Network flow information is the foundational knowledge for wireless networks. It encompasses sensitive information about the network, such as the data rates at which data flows between source-destination pairs along end-to-end paths. If malicious adversaries possess such knowledge, they can understand who is communicating with whom or the data rate between two communicating parties, and then launch effective attacks against the network [? ?]. For example, given the flow pattern, attackers can create profiles of individual devices or users based on their behavior. This can assist in targeting specific devices for attacks or tailoring phishing attempts to match the behavior of particular users. The direct observation of end-to-end flow information in some wireless networks, such as wireless sensor networks (WSNs) [? ?] and mobile ad-hoc networks (MANETs) [? ?], often remains unattainable or could be prohibited due to various reasons such as privacy concerns, legal restrictions, or technical limitations[? ? ? ? ?]. For example, in MANETs, nodes communicate directly with each other, forming a self-organizing network without a fixed centralized infrastructure to control the routing flows, making monitoring them challenging.

Network inference, also known as network tomography [? ? ? ?], is designed as a process of indirectly inferring sensitive flow information by observing link metrics that are easy to capture in a wireless network[? ? ? ? ? ? ? ? ? ? ? ?]. Network inference involves using relationships between end-to-end flow rates and link rates, often determined by the routing protocols and network topology, to make inferences. However, when malicious attackers leverage network inference to analyze and infer flow information from easy-captured and seemingly innocuous data, they can uncover valuable information that can aid in their attacks [? ? ? ? ? ? ?]. Network inference eliminates the requirement for attackers to gain access to the network, which results in significant security and privacy concerns.

For successful network inference, attackers should rely on consistent patterns and behavior. Therefore, dynamic routing can be used to mitigate wireless inference attacks. Dynamic routing protocols continually adjust the paths that data packets take within a wireless network [? ? ? ? ?]. This means that even if an attacker is monitoring network traffic over time, the flow rates and communication paths they observe will be constantly changing, creating mismatch between the observed traffic pattern and the actual pattern used in the network. This mismatch disrupts the attacker's ability to derive an accurate inference from the measured link information. In literature, many dynamic routing protocols have been proposed. In our early work [**?**], the performance of many existing dynamic routing protocols was investigated and compared without proposing new protocols. However, we notice that most of existing protocols prioritize security objectives other than defending network inferences, thereby hindering resilience performance. For example, in Tor [**?**], anonymous communication is achieved by randomly choosing three relays; however, it remains unclear whether the randomness of Tor is sufficient to prevent network inference attack. In this paper, we observe that the inference error is positively related to the probability of the mismatch between the flow template, which is characterized as the random matrix in network inference, observed by the attackers, and the real template used in the network. Motivated by this, we propose a dynamic routing protocol, called Max-Mismatch-Probability (MMP), which seeks to maximize mismatch probability and increase the inference error. We conduct a comprehensive theoretical analysis to demonstrate that our proposed method MMP can achieve the inference error on the same order of the number of nodes in the network.

Following is a summary of the contribution of this paper.

- We design a routing protocol that specifically addresses the resilience of the network inference attack on wireless networks, and then we propose a protocol called Max-Mismatch-Probability to prevent this attack.
- We present a comprehensive theoretical analysis of the performance of MMP against inference attacks, i.e., inference errors, and the cost of the protocol, which is the delay. Using MMP, we have shown that flow information can be concealed by inference errors.
- Simulations are conducted to demonstrate the performance and cost of MMP, and the results verify the theoretical analysis.

The remainder of this paper is organized as follows. In Section 2, we briefly introduce the network inference and state our problem. In Section 3, we introduce our mathematical model of the routing protocol and propose our design. In Section 4 we provide theoretical analysis and deliver and prove our results. In Section 5, we use simulation experience to verify our results. Finally, we present related work in Section 6 and conclude in Section 7.

2 ATTACK MODELING AND DESIGN MOTIVATION

In this section, we first present the network model and the background of network inference. Then, we state our research problems. All notations are defined in Table **??**. Without extra specification, in this paper, the upper-case bold indicates a matrix, the lower-case bold indicates a vector, and the calligraphy font indicates a set.

2.1 Wireless Network Model

The topology of a wireless network is modeled as a random geometric graph (RGG), which has been widely used for modeling distributed wireless networks [**?**]. Denoted by $\mathcal{G} = (\mathcal{V}, \mathcal{L})$ the RGG, where $\mathcal{V}$ is the node set and $\mathcal{L}$ is the undirected link set. Let $N = |\mathcal{V}|$ and $L = |\mathcal{L}|$ be the total number of nodes and links respectively. In this network, each node represents an RF end, and N nodes are randomly placed in a region $\Omega = [0, \sqrt{N/\lambda}]^2$, where λ denotes the node density, and we assume λ is sufficiently large such that the entire network is connected asymptotically almost surely [**?**]. Denoted by r the transmission range of each node, and two nodes are connected if they are in each other's transmission range. Note that adopting whether the directed or undirected case has no impact on our formulation of our problem and the directed case is a straightforward extension of the undirected case.

Table 1: Notations.

$\mathbf{X}^T$	The transpose of matrix $\mathbf{X}$.
$\mathbf{X}^{-1}$	The inverse of matrix $\mathbf{X}$.
$\|\mathbf{x}\|_p$	The $\mathcal{L}$-p norm of vector $\mathbf{x} = [x_1, x_2, \cdots, x_n]^T$.
$f(n) = O(g(n))$ or $g(n) = \Omega(f(n))$	$\exists n_0$, there exists a constant c such that $f(n) \leq cg(n)$ for $\forall n > n_0$.
$f(n) = \Theta(g(n))$	$f(n) = O(g(n))$ and $f(n) = \Omega(g(n))$.
$\|\mathcal{F}\|$	The cardinality of set $\mathcal{F}$.
$\mathbf{tr}\{\mathbf{X}\}$	the trace of $\mathbf{X}$.
$\lfloor x \rfloor$	the floor of a scalar x.

In the network, packets are exchanged between nodes in a node pair, resulting in 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. Therefore, there are $|\mathcal{F}| = N(N-1)/2$ flows in this network, which is also the number of node pairs. Denoted by x_i the data rate of flow $f_i \in \mathcal{F}$, then we have 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). By analyzing the flow rate vector $\mathbf{x}$, we can determine who is communicating with whom in the network and how much data rate they have. The disclosure of such information is undesirable or even prohibited in many practical scenarios such as military and civil applications [**? ? ?**].

2.2 Wireless Network Inference Attacks

The flow rate vector $\mathbf{x}$ contains important information for the network. Malicious adversaries can launch powerful, effective attacks against a network when they possess such information. As a result, we model the attacker's objective as obtaining the flow rate vector $\mathbf{x}$. It should be noted that although each link is connected wirelessly with a broadcast nature, the flow rate vector $\mathbf{x}$ is usually not directly measurable by the attacker since the flow information is indicated at the network or higher layers[**?**], whose data is typically encrypted at the physical or link layers. As a result, the attacker has to indirectly infer this information from physical and link-layer activities, which is referred to as network inference.

Mathematically, let a column vector $\mathbf{y} = [y_1, y_2, \cdots, y_L]^T$ be a measured link rate vector where y_i denotes the rate of the link l_i. In network inference, the relationship between the link rate vector $\mathbf{y}$ and the flow rate vector $\mathbf{x}$ can be modeled as the following linear system

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

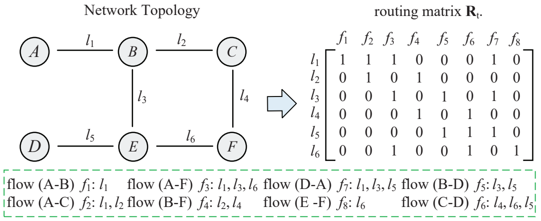
Network Topology
routing matrix $\mathbf{R}_t$.
A
B
C
D
E
F
l_1
l_2
l_3
l_4
l_5
l_6
f_1 f_2 f_3 f_4 f_5 f_6 f_7 f_8
l_1 1 1 1 0 0 0 1 0
l_2 0 1 0 1 0 0 0 0
l_3 0 0 1 0 1 0 1 0
l_4 0 0 0 1 0 1 0 0
l_5 0 0 0 0 1 1 1 0
l_6 0 0 1 0 0 1 0 1
flow (A-B) f_1: l_1
flow (A-F) f_3: l_1, l_3, l_6
flow (D-A) f_7: l_1, l_3, l_5
flow (B-D) f_5: l_3, l_5
flow (A-C) f_2: l_1, l_2
flow (B-F) f_4: l_2, l_4
flow (E -F) f_8: l_6
flow (C-D) f_6: l_4, l_6, l_5

Figure 1: Example of a network consisting of 6 nodes (nodes $A \cdots F$), 6 links (links $l_1 \cdots l_6$) and 8 flows (flows $f_1 \cdots f_8$), and a routing matrix R built based on the shortest-path routing protocol.

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

$$r_{ij} = \begin{cases} 1, & \text{if link } l_i \text{ is present on a path of flow } f_j \text{ ;} \\ 0, & \text{otherwise.} \end{cases} \tag{2}$$

Information in $\mathbf{R}$ illustrates how flows are constructed by links based on routing protocols. Denoted by T the routing protocol used in a network. Figure **??** demonstrates a toy example showing how the routing matrix is built based on the shortest-path routing protocol. This network consists of 6 nodes, 6 links, and 8 flows. Therefore the link set $\mathcal{L} = \{l_1, \cdots, l_6\}$ and flow set $\mathcal{F} = \{f_1, \cdots, f_8\}$.

Assume the shortest-path protocol is implemented in this network, and we define the number of hops as the distance, then each flow will select the shortest path for routing packets. For example, the flow f_2 will go through links l_1 and l_2 which have the shortest distance 2, and the distance of all other paths are longer than this path, e.g., the distance of path $A \longrightarrow B \longrightarrow E \longrightarrow F \longrightarrow C$ is 4, thus we avoid using it. Then the second column of the routing matrix $\mathbf{R}$ is $[1\ 1\ 0\ 0\ 0\ 0]^T$ indicating that flow f_2 is concatenated by l_1 and l_2.

In a wireless network, it is easy to obtain the link rate vector $\mathbf{y}$ through eavesdropping. The routing matrix $\mathbf{R}$ is determined by the network topology and routing protocol. In this paper, we consider a powerful attacker who has an entire knowledge of the network topology and routing protocol. Denoted by $\hat{\mathbf{x}_t}$ the inferred value of flow rate vector $\mathbf{x}_t$, and defines the inference error ϵ as

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

Then we can model the objective of the attacker is to obtain an inferred flow rate vector $\hat{\mathbf{x}}$ which has the minimized inference error ϵ, given the knowledge of the link rate vector y, the network topology $\mathcal{G}$, the routing protocol T, i.e.,

$$\begin{aligned} \text{Objective} : \hat{\mathbf{x}} &= \arg\min \epsilon \\ \text{Given} : \mathbf{y}_t&,\ \mathcal{G},\ T. \end{aligned} \tag{4}$$

In a wireless network, the number of flows $|\mathcal{F}|$ is usually larger than the number of links L, resulting in the linear system (**??**) under-determined, thus the attacker can leverage any optimization algorithms to minimize the inference error ϵ. In Figure **??**, the shortest-path protocol is used, then we have the routing path of each traffic flow will be fixed onto the one with the shortest distance. If flow rate of f_4 is 10bps and other flows have no data exchange, i.e., $\mathbf{y} = [0\ 0\ 0\ 10\ 0\ 0\ 0\ 0\ ]$, and given the routing matrix $\mathbf{R}$, we can know the rate of links l_2 and l_4 are 10bps, i.e., $\mathbf{x} = [0\ 10\ 0\ 10\ 0\ 0\ ]$.

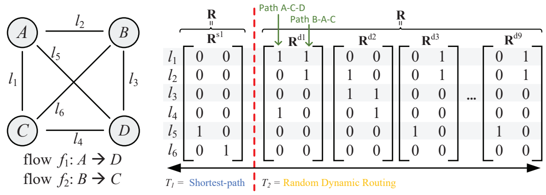
A
B
C
D
l_1
l_2
l_3
l_4
l_5
l_6
flow f_1: $A \rightarrow D$
flow f_2: $B \rightarrow C$
Path A-C-D
Path B-A-C
$\mathbf{R} = \mathbf{R}^{s1}$
$\mathbf{R} = \mathbf{R}^{d1}$ $\mathbf{R}^{d2}$ $\mathbf{R}^{d3}$ ... $\mathbf{R}^{d9}$
l_1 0 0 | 1 1 | 0 0 | 0 1 | 0 1
l_2 0 0 | 0 1 | 1 0 | 0 1 | 0 1
l_3 0 0 | 0 0 | 1 1 | 0 0 | 0 0
l_4 0 0 | 1 0 | 0 1 | 0 0 | 0 0
l_5 1 0 | 0 0 | 0 0 | 1 0 | 1 0
l_6 0 1 | 0 0 | 0 0 | 0 0 | 0 0
T_1 = Shortest-path
T_2 = Random Dynamic Routing

Figure 2: Illustrative example of the difference of routing matrix R in static routing (e.g., the shortest-path) and a random static routing protocol.

In this work, we assume the optimization algorithm used by the attacker is agnostic to us which is the worst case since if we know it, we may provide a easier method-specific defense strategy.

2.3 Dynamic Routing

Taking a close look at (**??**), the inference error will be induced by two factors: the routing matrix $\mathbf{R}$, and the eavesdropping link rate vector $\mathbf{y}$. Since the wireless medium is open, the attacker is always able to acquire an accurate link rate vector $\mathbf{y}$ through advanced spectrum sniffing techniques [**? ?**], thus our focus in this work is on investigating how the routing matrix $\mathbf{R}$ can affect the attacker for inferring the flow rate.

Even though the routing matrix $\mathbf{R}$ is not directly available to the attacker in some network scenarios, the attacker is able to construct it based on the routing protocol T and network topology $\mathcal{G}$. Denoted by $\hat{\mathbf{R}}$ the constructed routing matrix by the attacker. Assuming the optimization algorithm is accurate enough, then the inference error ϵ will be solely dominated by the mismatch between the constructed routing matrix $\hat{\mathbf{R}}$ and the real matrix $\mathbf{R}$. In the case of static routing, such as the shortest-path, the routing matrix $\mathbf{R}$ is uniquely constructed based on the routing protocol T and will be fixed for all communication rounds. Even in the scenario where routing protocol is not available, the attacker is still able to obtain it through sensing. Therefore The attacker can build the exact routing matrix with high probability, i.e., $\hat{\mathbf{R}} = \mathbf{R}$, resulting in an accurate inference of the flow rate vector $\hat{\mathbf{x}}$. In contrast, the routing path of each flow changes at each communication round under a dynamic routing protocol. Given the routing protocol T and topology $\mathcal{G}$, the routing matrix $\mathbf{R}$ cannot be uniquely determined, making constructing an accurate routing matrix extremely difficult, resulting in more inference errors to the attacker than deterministic ones. Previous work in [**?**] analyzed the performance of many existing dynamic routing protocols without proposing new protocols. However, we notice that most of existing dynamic protocols prioritize security objectives other than defending network inferences, thereby hindering resilience performance. Therefore, in this work, we are interest in designing a routing protocol that specifically addresses the resilience of the network inference attack in wireless networks.

In Figure **??**, an illustrative example is provided of how the routing matrix $\mathbf{R}$ in a 4-node, 6-link network is constructed based on two kinds of routing strategies, i.e., T_1: shortest-path and T_2: random dynamic routing. In this example, there are two traffic flows, i.e., flow f_1 from node A to D, and flow f_2 from node B to C, in

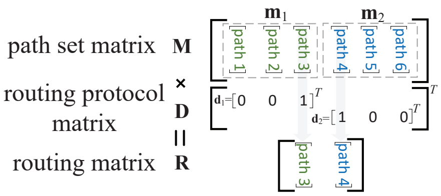

Figure 3: Example of how the routing matrix R_t can be decomposed into the path set matrix M and the routing protocol matrix D_t.

which each flow contains 3 potential paths, i.e., the flow f_1 may use path $l_1 \rightarrow l_4$, or $l_2 \rightarrow l_3$ or l_5, and flow f_2 may use path $l_2 \rightarrow l_1$, or $l_3 \rightarrow l_4$ or l_6. For the shortest-path protocol T_1, the routing matrix can be uniquely determined. Flow f_1 only goes through link l_5 and flow f_2 goes through link l_6, thus there is only one candidate $\mathbf{R}^{s1}$ that can be selected by the routing matrix. Accordingly, by using the shortest-path protocol, it is very likely to construct a routing matrix $\hat{\mathbf{R}}$ by the attacker such that $\hat{\mathbf{R}} = \mathbf{R}$, from which an accurate flow rate vector can be derived. However, for the dynamic routing strategy T_2, where the routing matrix can be randomly selected from $3 \times 3 = 9$ candidates (e.g., $\mathbf{R}^{d1}, \cdots, \mathbf{R}^{d9}$) at each transmission round. As the routing matrix selection is random and unpredictable, even if the attacker is aware of the routing protocol, the attacker cannot gain any priority on the selection of the current routing matrix, and the probability that $\hat{\mathbf{R}} = \mathbf{R}$ is $1/9$. Therefore, the matrix mismatch will introduce additional errors, which, in turn, will protect the network against inference attacks.

3 ROUTING PROTOCOL MODELING AND DESIGN

In Figure **??**, we demonstrate that introducing additional errors is the basic idea for why dynamic routing can protect networks against network inference attacks. Therefore, this paper aims to propose a dynamic routing protocol T that maximizes the inference error ϵ. In the following, we first model the dynamic routing protocol and then propose our protocol.

3.1 Routing Protocol Decomposition and Modeling

Each traffic flow can include multiple paths in a wireless network, and the routing protocol determines which path is selected. For example in Figure **??**, there are two traffic flows and each flow includes 3 paths, therefore, the routing protocol will choose one path for each flow and totally there are 9 combinations. To have a better understanding of the routing protocol, according to [**?**], we decomposite the routing matrix $\mathbf{R}$ into two parts, i.e., $\mathbf{R} = \mathbf{M} \times \mathbf{D}$, where $\mathbf{M}$ is the path set matrix and $\mathbf{D}$ is the routing protocol matrix.

The path set matrix $\mathbf{M} = [\mathbf{m}_1, \mathbf{m}_2, \cdots, \mathbf{m}_F]$ represents all paths of every traffic flow in the network, in which each entry $\mathbf{m}_i$ indicates the path set of flow f_i. Denoted by $\mathcal{P}_i$ the path set of flow $f_i \in \mathcal{F}$, and let $\mathcal{P} = \{\mathcal{P}_i\}_{i=1,\cdots,F}$ be path set of every flow in the network. Define a function V to map paths to column vectors in the routing matrix, then we have $V(\mathcal{P}_i) = \mathbf{m}_i$ and $V(\mathcal{P}) = \mathbf{M}$. Note that a path set matrix $\mathbf{M}$ can be uniquely determined when a specific topology is provided.

The routing protocol matrix $\mathbf{D} = \text{diag}(\mathbf{d}_1, \mathbf{d}_2, \cdots, \mathbf{d}_F)$ is responsible for modeling the behavior of a routing protocol into a matrix that selects a particular path for each active flow at each communication round. Every $\mathbf{d}_i$ is a column vector with length $|\mathcal{P}_i|$, in which only one entry is 1 and the remaining entries are all 0, identifying the selected path. For the static routing protocol $\mathbf{D}$ is unique, rendering the routing matrix $\mathbf{R}$ unique as well. For the dynamic routing protocol, the $\mathbf{D}$ changes over time, making the routing matrix $\mathbf{R}$ dynamic.

Figure **??** shows how the decomposition happens in the example of Figure **??**. Since there are two active traffic flows and each flow contains 3 paths. Thereby, we have the path set matrix $\mathbf{M} = [\mathbf{m}_1, \mathbf{m}_2]$, and each m_1 includes three columns indicating 3 potential paths. We assume the routing protocol selects path 3 and path 4 for flow 1 and flow 2, respectively. Then we can see $\mathbf{d}_1$ is $[0, 0, 1]^T$ (i.e., select path 3) and $\mathbf{d}_2$ is $[1, 0, 0]^T$ (i.e., select path 4). Then the routing matrix is constructed by path 3 and path 4.

Based on the decomposition $\mathbf{R} = \mathbf{MD}$ we can model the routing protocol T as follows.

Model 1. *[Routing Protocol] In a network, the routing protocol can be modeled as a function T to derive the routing protocol matrix $\mathbf{D}$, given the network topology $\mathcal{G}$, the path set matrix $\mathbf{M}$, i.e.,*

$$D_t = T(\mathcal{G}, \mathbf{M}). \tag{5}$$

Remark 1. *The Model **??** divides the construction of the routing matrix $\mathbf{R}$ into two separate factors i.e., the path set matrix $\mathbf{M}$ and the routing protocol matrix $\mathbf{D}$, and build the connection between the routing protocol T and the routing protocol matrix $\mathbf{D}$. The static routing protocol can be considered as a one-to-one mapping, thus given the input, the routing protocol matrix $\mathbf{D}$ can be uniquely derived. In contrast, the dynamic routing protocol can be considered as a one-to-many mapping, which derives a set consisting of many candidates of routing protocol matrices. The probability that the attacker has a correct construction depends on the size of the set.*

3.2 Max-Mismatch-Probability Routing

The dynamic routing protocol builds a set of routing protocol matrices. For example, in Figure **??**, there are 9 candidate matrices. At each communication round, the network randomly chooses one $\mathbf{D}$ to determine the routing matrix $\mathbf{R}$, and the attacker will also randomly choose one $\hat{\mathbf{D}}$ to construct $\hat{\mathbf{R}}$ and then launch the inference attack. The selection mismatch, i.e., $\hat{\mathbf{D}} \neq \mathbf{D}$, will cause the attacker and the network to construct different routing matrices, i.e., $\hat{\mathbf{R}} \neq \mathbf{R}$, and eventually induce inference error. To maximize the inference error, the intuitive strategy is to select matrix $\mathbf{D}$ that has the maximum difference $\|\hat{\mathbf{R}} - \mathbf{R}\|$. However, this strategy requires knowing the matrix selected by the attacker $\hat{\mathbf{D}}$, which is usually unavailable.

Instead, we notice that, in a random graph, the inference error is also related to the probability that the attacker and the network will choose different routing protocol matrices, and this probability is determined by the number of candidate matrices generated by the routing protocol. For example, if there is only one candidate matrix, then the probability of choosing a different matrix will be 0. For the example in Figure ??, there are 9 candidate matrices therefore the probability of mismatch is $1 - 1/9$. The number of candidate matrices depends on the number of paths for each flow and the routing protocol. Assuming that a topology exists, the number of paths for each flow will be fixed, then the protocol will determine how large the search space will be. For example, Tor will select three relays randomly for tuning the path, thus only a subset of paths that pass through those relays are considered. It is easy to know, the maximum mismatch probability is $1 - 1/\prod \mathcal{P}_i$, therefore, we propose a dynamic routing protocol designed to achieve this probability. We call this protocol Max-Mismatch-Probability (MMP) routing. Specifically, at each communication round, each packet in a traffic flow f_i is transmitted through one path that is randomly selected from all $\mathcal{P}_i$. In this way, the maximum probability can be achieved.

4 THEORETICAL ANALYSIS

In this section, we provide the theoretical analysis of our proposed protocol for the upper and lower bound of the interference error. We first introduce the genie bound and then we show our theoretical results.

4.1 Genie Bound

As aforementioned, the attacker can leverage any optimization algorithm to infer the flow rate vector $\hat{\mathbf{x}}$, and we are agnostic to optimization algorithm. To remove the impact from the algorithm, a commonly used method is to leverage the genie bound [?] to measure the inference error ϵ.

Specifically, 1) we first construct a new flow rate vector $\mathbf{x}_g$ via deleting zero entries by the help of a genie, and $\mathbf{R}_g$ that is the routing matrix based on $\mathbf{x}_g$. This step shrinks the system by removing the flows with no data traffic and have a new system

$$\mathbf{y} = \mathbf{R}_g \mathbf{x}_g. \tag{6}$$

2) Use the least square to derive the estimation of $\mathbf{x}_g$, i.e.,

$$\hat{\mathbf{x}}_g = (\mathbf{R}_g^T \mathbf{R}_g)^{-1} \mathbf{R}_g^T \mathbf{y}. \tag{7}$$

3) Finally, the genie bound can be derived as the mean square error of $\hat{\mathbf{x}}_g$ and $\mathbf{x}_g$

$$G(\mathbf{x}_g) = \mathbb{E}\left(||\hat{\mathbf{x}}_g - \mathbf{x}_g||_2^2| \right). \tag{8}$$

In solving under-determined systems, the genie bound is widely used to provide a lower error bound regardless of the method of inference used. Using the genie bound, we redefine the inference error as

$$\epsilon_g = \mathbb{E}\left(||\hat{\mathbf{x}}_g - \mathbf{x}_g||_2^2| \right) \tag{9}$$

Remark 2. *The first two steps convert the under-determined system* (??) *to a determined system by eliminating nonexistent flows, so that inference approaches have no effect on deriving the genie bound. Then the inference error* ϵ *in* (??) *is measured only on the flows with real traffics, serving as a general, method-independent error bound.*

Remark 3. *Note that the estimation* (??) *is available under the condition that the row rank of* $\mathbf{R}_g$ *is no less than the column rank of* $\mathbf{R}_g$. *This condition is always valid because* $\mathbf{x}$ *is spare, and in practice, it is less likely that every node is communicating with others. Denoted by* $\mathcal{F}_g$ *the corresponding flow set of* $\mathbf{x}_g$, *obtained by removing nonexisting flows from* $\mathcal{F}$. *In the rest of this paper, we assume* $L > F$ *where* $F = |\mathcal{F}_g|$, *thereby this condition always holds.*

4.2 Theoretical Results

Consider the network $\mathcal{G}$ with N nodes, F traffic flows. Assume the flow rate $x_i \in \mathbf{x}_g$ is a random variable with mean μ and variance σ^2. Then we have the following theorems.

Theorem 1. *[Inference Error] For the proposed MMP routing protocol, the inference error* ϵ_g *satisfies*

$$\Theta\left(\frac{F^2\mu^2(N-1)^2}{N^2(\sqrt{(N)}+F)}\right) \leq \epsilon_g \leq \Theta\left(\frac{2F^2(\mu^2+\sigma^2)}{N/(N-1)}\right). \tag{10}$$

Remark 4. *Results showing in the Theorem* ?? *indicates the inference error is affected by the number of flows F in a network. Then if we increase the number of data flows in the network to make the communication scenario in the network more complicated and intensive, the inference error increases quadratically. In addition, the traffic with larger data rate will also induce more inference error than the slow rate traffic.*

Compared to the shortest-path protocol, the proposed MMP protocol incurs a large inference error. However, using a random path for communication will cause a longer delay, since packets are not always routed through the shortest route. In order to measure the extra cost due to the MMP rounding protocol, we use the distance between two nodes as the metric of the delay. Denoted by τ the average distance of all node pairs, then we have the following theorem.

Theorem 2. *[Delay] The delays of MMP satisfy*

$$\tau = \Theta(\sqrt{N}). \tag{11}$$

Remark 5. *Theorem* ?? *shows that the average delay is on the order of* $\sqrt{N}$ *which shares the same order with the typical shortest-path protocol. This indicates that MMP routing protocol does not induce significant delay increase comparing with the shortest-path protocol, however the security enhancement is significant against inference attacks.*

4.3 Theorem Proof

We first prove Theorem ?? and then prove Theorem ?? because the results from Theorem ?? will be used for the proof of Theorem ??.

4.3.1 *Proof of Theorem* ??.

In a network $\mathcal{G}$, let the distance of $\forall f_i \in \mathcal{F}$ is d_i i.e., the number of hops of the shortest path. According to [?], if the density λ is large enough such that the all nodes in the network is connected, then the expected number of path for a node pair is $\Theta(N)$. Then the average distance of flow f_i can be expressed as

$$\tau_{f_i} = d_i + \frac{1}{sN}\sum_{j=1}^{k} c_{ij}, \tag{12}$$

where s is an arbitrary positive scalar and c_{ij} is a positive constant showing the distance difference between path $p_j \in \mathcal{P}_i$ to the shortest path. Then the average distance in the network can be expressed as

$$\tau = \frac{1}{F}\sum_{i=1}^{F} d_i + \frac{1}{FsN}\sum_{i=1}^{F}\sum_{j=1}^{sN} c_{ij}. \tag{13}$$

According to Lemma **??**, we have

$$\frac{1}{F}\sum_{i=1}^{F} d_i = \Theta(\sqrt{N}) \tag{14}$$

Furthermore, we know

$$\frac{1}{FsN}\sum_{i=1}^{F}\sum_{j=1}^{sN} c_{ij} = \Theta(1) \tag{15}$$

then we can obtain $\tau = \Theta(\sqrt{N})$, and finish the proof. □

4.3.2 Proof of Theorem **??**. According to the Model **??**, then (**??**) can be rewritten by

$$\mathbf{y} = \mathbf{M}\mathbf{D}\mathbf{x}_g. \tag{16}$$

The MMP routing protocol randomly select a path for each flow $f_i \in \mathcal{F}$. Therefore, the routing protocol used by the attacker may differ from the network. Denoted by $\hat{\mathbf{D}}$ the matrix used by the attacker. Therefore, for the attacker, (**??**) can be expressed as $\mathbf{y} = \mathbf{M}\hat{\mathbf{D}}\mathbf{x}_g$. Then the inferred flow rate vector $\hat{\mathbf{x}}$ can be written as

$$\hat{\mathbf{x}}_g = [(\mathbf{M}\mathbf{D})^T\mathbf{R}\mathbf{D}]^{-1}(\mathbf{R}\mathbf{D})^T\mathbf{y}. \tag{17}$$

In the following, the proof logic is partially based on [**?**]. Then according to the genie bound (**??**), we have that

$$\begin{aligned}
G(\mathbf{x}_g) &= \mathbb{E}\left(\|\hat{\mathbf{x}}_g - \mathbf{x}_g\|_2^2\right) \\
&= \mathbb{E}\left(\|[(\mathbf{M}\hat{\mathbf{D}})^T\mathbf{M}\hat{\mathbf{D}}]^{-1}\mathbf{M}\hat{\mathbf{D}}^T\mathbf{y} - \mathbf{x}_g\|_2^2\right) \\
&= \mathbb{E}\left(\|[(\mathbf{M}\hat{\mathbf{D}})^T\mathbf{M}\hat{\mathbf{D}}]^{-1}(\mathbf{M}\hat{\mathbf{D}})^T(\mathbf{M}\mathbf{D}\mathbf{x}_g) - \mathbf{x}_g\|_2^2\right) \\
&= \mathbb{E}\left(\|\mathbf{B}[\mathbf{M}\mathbf{D} - \mathbf{M}\hat{\mathbf{D}}]\mathbf{x}_g\|_2^2\right) \\
&= \mathbb{E}\left(\|\mathbf{B}\Delta\mathbf{x}_g\|_2^2\right)
\end{aligned} \tag{18}$$

where $\mathbf{B} = [(\mathbf{M}\hat{\mathbf{D}})^T\mathbf{M}\hat{\mathbf{D}}]^{-1}\mathbf{M}\hat{\mathbf{D}}^T$, and $\Delta = \mathbf{M}\mathbf{D} - \mathbf{M}\hat{\mathbf{D}}$. In (**??**), Δ shows the mismatch between the constructed routing matrix by the attack and the real routing matrix, and a large mismatch is expected to induce a large inference error. According to Lemma **??**, we have the following relationship

$$\lambda_{\min}(\mathbf{B}^T\mathbf{B})\|\Delta\mathbf{x}_g\|_2^2 \le \|\mathbf{B}\Delta\mathbf{x}_g\|_2^2 \le \lambda_{\max}(\mathbf{B}^T\mathbf{B})\|\Delta\mathbf{x}_g\|_2^2, \tag{19}$$

where $\lambda_{\min}(\mathbf{B}^T\mathbf{B})$ and $\lambda_{\max}(\mathbf{B}^T\mathbf{B})$ denotes the minimum and maximum eigenvalues of $\mathbf{B}^T\mathbf{B}$. According to Lemma **??**, $\lambda_{\min}(\mathbf{B}^T\mathbf{B})$ and $\lambda_{\max}(\mathbf{B}^T\mathbf{B})$ can be replaced by $\lambda_{\max}^{-1}(\mathbf{M}\hat{\mathbf{D}}(\mathbf{M}\hat{\mathbf{D}})^T)$ and $\lambda_{\min}^{-1}(\mathbf{M}\hat{\mathbf{D}}(\mathbf{M}\hat{\mathbf{D}})^T)$ respectively, then (**??**) can be rewritten as

$$\mathbb{E}\left(\frac{\|\Delta\mathbf{x}_g\|_2^2}{\lambda_{\max}(\mathbf{M}\hat{\mathbf{D}}(\mathbf{M}\hat{\mathbf{D}})^T)}\right) \le G(\mathbf{x}_g) \le \mathbb{E}\left(\frac{\|\Delta\mathbf{x}_g\|_2^2}{\lambda_{\min}(\mathbf{M}\hat{\mathbf{D}}(\mathbf{M}\hat{\mathbf{D}})^T)}\right), \tag{20}$$

Next, we proceed to derive $\lambda_{\max}^{-1}(\mathbf{M}\hat{\mathbf{D}}(\mathbf{M}\hat{\mathbf{D}})^T)$ and $\lambda_{\min}^{-1}(\mathbf{M}\hat{\mathbf{D}}(\mathbf{M}\hat{\mathbf{D}})^T)$. According to Lemma **??**, we have that $\lambda_{\min}(\mathbf{M}\hat{\mathbf{D}}(\mathbf{M}\hat{\mathbf{D}})^T) = \Theta(\tau(N))$ and $\lambda_{\max}(\mathbf{M}\hat{\mathbf{D}}(\mathbf{M}\hat{\mathbf{D}})^T) \le \Theta\left(\tau(N) + F\tau^2(N)/N\right)$. Then the genie bound can be expressed as the following asymptotically solution

$$\frac{\mathbb{E}\|\Delta\mathbf{x}_g\|_2^2}{\Theta\left(\tau(N) + \frac{F\tau^2(N)}{N}\right)} \le \epsilon_g \le \frac{\mathbb{E}\|\Delta\mathbf{x}_g\|_2^2}{\Theta(\tau(N))}. \tag{21}$$

Next, we will derive the $\mathbb{E}\|\Delta\mathbf{x}_g\|_2^2$. Let each entry in Δ as δ_{ij}, where $i \in [1, L]$ and $j \in [1, F]$, and $\delta_{ij} \in \{0, 1, -1\}$. Since both the attacker and the network randomly select their paths for each flow, denoted by $g_j = |\mathcal{P}_j|$ the number of paths of flow f_j, then we have that

$$\begin{aligned}
\Pr\{\delta_{ij} = 1\} =& \Pr\{\delta_{ij} = 1|\mathbf{d}_j \ne \hat{\mathbf{d}}_j\}\Pr\{\mathbf{d}_j \ne \hat{\mathbf{d}}_j\} \\
=& \frac{g_j - 1}{g_j}\Pr\{\delta_{ij} = 1|\mathbf{c}_j \ne \mathbf{d}_j\} \\
=& \frac{g_j - 1}{g_j}\Theta\left(\frac{\tau(N)}{N}\right)\left(1 - \Theta\left(\frac{\tau(N)}{N}\right)\right).
\end{aligned} \tag{22}$$

Since the path selection is random, we know that $\Pr\{\delta_{ij} = -1\} = \Pr\{\delta_{ij} = 1\}$. All nodes are placed uniformly, thus according to [**?**], all flows will have the same expected number of paths, i.e., $\mathbb{E}(g_j) = g = \Theta(N)$. Then we have

$$\begin{aligned}
\mathbb{E}\{\delta_{ij}\} &= \Pr\{\delta_{ij} = 1\} - \Pr\{\delta_{ij} = -1\} \\
&= \Theta\left(\frac{N-1}{N} \times \frac{\tau(N)}{N}\right) \\
&= \Theta\left(\frac{(N-1)\tau(N)}{N^2}\right),
\end{aligned} \tag{23}$$

and

$$\begin{aligned}
\mathbb{E}\{\delta_{ij}^2\} &= \Pr\{\delta_{ij} = 1\} + \Pr\{\delta_{ij} = -1\} \\
&= \Theta\left(\frac{2(N-1)}{N} \times \frac{\tau(N)}{N}\right) \\
&= \Theta\left(\frac{2(N-1)\tau(N)}{N^2}\right).
\end{aligned} \tag{24}$$

Then the lower bound and upper bound of $\mathbb{E}\|\Delta\mathbf{x}_g\|_2^2$ can be derived as follows.

$$\begin{aligned}
\mathbb{E}\|\Delta\mathbf{x}_g\|_2^2 &= \mathbb{E}\left(\sum_{i=1}^{L}\left(\sum_{j=1}^{F}\delta_{ij}x_j\right)^2\right) \\
&= \Theta(N)\mathbb{E}\left(\left(\sum_{j=1}^{F}\delta_{ij}x_j\right)^2\right) \\
&\ge \Theta(N)\left(\mathbb{E}\left(\sum_{j=1}^{F}\delta_{ij}x_j\right)\right)^2 \\
&= \Theta\left(\frac{[F\mu(N-1)\tau(N)]^2}{N^3}\right).
\end{aligned} \tag{25}$$

and by leveraging Cauchy-Schwarz inequality

$$\begin{aligned}\mathbb{E}\|\Delta \mathbf{x}_g\|_2^2 &= \Theta(N)\mathbb{E}\left(\left(\sum_{j=1}^{F} \delta_{ij}x_j\right)^2\right)\\ &\le \Theta(N)\mathbb{E}\left(\sum_{j=1}^{F} \delta_{ij}^2\right)\mathbb{E}\left(\sum_{j=1}^{F} x_j^2\right)\\ &= \Theta\left(\frac{2F^2(N-1)(\mu^2+\sigma^2)\tau(N)}{N}\right).\end{aligned} \tag{26}$$

Replacing the result of $\tau(N)$ from Theorem **??**, and (**??**), (**??**) into (**??**), we can complete the proof. □

4.3.3 Proof of Lemmas.

Lemma 1. *For network $\mathcal{G}$ consisting of N nodes, for all path based on the shortest path protocol we have the average distance as $d = \frac{1}{F}\sum_{i=1}^{F} d_i = \Theta(\sqrt{N})$, where d_i is the distance of arbitrary flow $f_i \in \mathcal{F}$.*

Proof: Denoted by e_i the Euclidean distance for flow f_i. The distance of each hop is $\Theta(r)$, thus the distance of flow f_i by using the shortest path 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

$$\frac{1}{N}\sum_{i=1}^{N} e_i = \Theta\left(\sqrt{N/\lambda}\right). \tag{27}$$

Therefore the average delay can be derived as

$$d = \Theta\left(\frac{\sqrt{N/\lambda}}{r}\right) = \Theta(\sqrt{N}), \tag{28}$$

which completes the proof. □

Lemma 2. *For an arbitrary matrix $\mathbf{B}$ and an arbitrary vector α, it satisfies*

$$\lambda_{\min}(\mathbf{B})\|\alpha\|_2^2 \le \|\mathbf{B}\alpha\|_2^2 \le \lambda_{\max}(\mathbf{B})\|\alpha\|_2^2, \tag{29}$$

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

Proof: It is easy to know

$$\|\mathbf{B}\alpha\|_2^2 = \|\alpha^T\mathbf{B}^T\mathbf{B}\alpha\| = \frac{\|\alpha^T\mathbf{B}^T\mathbf{B}\alpha\|}{\alpha^T\alpha}\|\alpha\|_2^2. \tag{30}$$

Based on [?], $\frac{\|\alpha^T\mathbf{B}^T\mathbf{B}\alpha\|}{\alpha^T\alpha}$ has the minimum bound $\lambda_{\min}(\mathbf{B}^T\mathbf{B})$ and the maximum bound $\lambda_{\max}(\mathbf{B}^T\mathbf{B})$, then we complete the proof. □

Lemma 3. *For a matrix $\mathbf{B} \in \mathbb{R}^{m\times n}$ where $m > n$, let $\mathbf{H} = (\mathbf{B}^T\mathbf{B})^{-1}\mathbf{B}^T$, then we have $\lambda_{\max}(\mathbf{H}^T\mathbf{H}) = \lambda_{\min}^{-1}(\mathbf{B}^T\mathbf{B})$ and $\lambda_{\min}(\mathbf{H}^T\mathbf{H}) = \lambda_{\max}^{-1}(\mathbf{B}^T\mathbf{B})$.*

Proof: According to singular value decomposition, for matrix $\mathbf{B} \in \mathbb{R}^{m\times n}$, then we have two unitary matrices: $\mathbf{V} \in \mathbb{R}^{n\times n}$ and $\mathbf{U} \in \mathbb{R}^{m\times m}$, such that $\mathbf{B} = \mathbf{U}\Lambda\mathbf{V}^T$ where $\Lambda = \mathrm{diag}(s_1, \cdots, s_n) \in \mathbb{R}^{m\times n}$ is a rectangular diagonal matrix, where $s_i = \lambda_i(\sqrt{\mathbf{B}^T\mathbf{B}})$ for $i = 1, \cdots, n$ are singular values of $\mathbf{B}$. Let $D := \mathrm{diag}(s_1^2, \cdots, s_n^2) \in \mathbb{R}^{n\times n}$, we have $\mathbf{B}^T\mathbf{B} = \mathbf{V}D\mathbf{V}^T$, thus

$$(\mathbf{B}^T\mathbf{B})^{-1} = \mathbf{V}D^{-1}\mathbf{V}^T. \tag{31}$$

Then we can derive $\mathbf{H}$ as

$$\mathbf{H} = (\mathbf{B}^T\mathbf{B})^{-1}\mathbf{B}^T = \mathbf{V}D^{-1}\mathbf{V}^T\mathbf{V}\Lambda\mathbf{U}^T = \mathbf{V}\Lambda^{-1}\mathbf{U}^T, \tag{32}$$

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

$$\mathbf{H}^T\mathbf{H} = \mathbf{U}(\Lambda^{-1})^2\mathbf{U}^T = \mathbf{U}D^{-1}\mathbf{U}^T. \tag{33}$$

Combining (**??**) with (**??**) we have that $\lambda(\mathbf{H}^T\mathbf{H}) = \lambda^{-1}(\mathbf{B}^T\mathbf{B})$, which completes the proof. □

Lemma 4. *For a random binary matrix $\mathbf{B} \in \mathbb{R}^{L\times F}$, let the expectation of each entry be $\mathbb{E}(b_{ij}) = \Theta(\tau(N)/N)$ with $\tau(N) = O(N)$ and $L = \Theta(N)$. Then if $F \to \infty$ with $\lim_{L\to\infty} F/L < \infty$, then the following statements are satisfied almost surely,*

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

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

Proof:

We first prove statement (1). According to [?], Let f be a function satisfying $\mathrm{Var}(f\mathbf{A}) = 1$. Then we have

$$\lambda_{\min}(\mathbf{B}^T\mathbf{B}) = \frac{L}{f^2}\lambda_{\min}(L^{-1}(c\mathbf{B})^T(c\mathbf{B})). \tag{34}$$

From [?], we have $\lambda_{\min}(L^{-1}(c\mathbf{B})^T(c\mathbf{B})) = \Theta(1)$ with high probability. Then according to [?], the number of links is on the order of nodes with high probability in an RGG, i.e., $L = \Theta(N)$ happens with high probability, then we know that

$$\lambda_{\min}(\mathbf{B}^T\mathbf{B}) = (L/f^2)\Theta(1) = \frac{\Theta(N)}{\Theta(N/\tau(N))} = \Theta(h(N)). \tag{35}$$

Now we prove statement (2). Let $\mathbf{U}$ and $\mathbf{V}$ be two matrix where $\mathbf{V}$ is an all-one matrix and each entry in $u_i \in \mathbf{U}$ satisfies $\mathbb{E}(u_i) = 0$, and $\mathbf{U}$ and $\mathbf{V}$ satisfies

$$\mathbf{B} = \mathbf{U} + \mathbf{V}h(N)/N. \tag{36}$$

Replacing into $\mathbf{B}^T\mathbf{B}$, then we have that

$$\begin{aligned}\lambda_{\max}(\mathbf{B}^T\mathbf{B}) &= \lambda_{\max}((\mathbf{U}^T + \mathbf{V}^T\tau(N)/N)(\mathbf{U} + \mathbf{V}\tau(N)/N))\\ &\le \lambda_{\max}(\mathbf{U}^T\mathbf{U}) + 2\tau(N)/N\lambda_{\max}(\mathbf{U}^T\mathbf{V})\\ &\quad + (\tau(N)/N)^2\lambda_{\max}(\mathbf{V}^T\mathbf{V}).\end{aligned} \tag{37}$$

Because the rank of $\mathbf{V}$ is 1, then we have that

$$\lambda_{\max}(\mathbf{U}^T\mathbf{V}) = \mathrm{tr}\{\mathbf{U}^T\mathbf{V}\} = \sum_i\sum_j u_{ij}. \tag{38}$$

According to the large number law,

$$\lambda_{\max}(\mathbf{U}^T\mathbf{V}) = o(NF)^{1/p} \tag{39}$$

for $1 < p < 2$. Then similarly, we have

$$\lambda_{\max}(\mathbf{V}^T\mathbf{V}) = \sum_{v_{ij}\in\mathbf{V}} 1 = \Theta(FN) \tag{40}$$

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

5 EXPERIMENTAL RESULTS

In this section, we demonstrate the performance of our proposed routing strategy MMP through simulation. In the following, we first introduce the experimental setups, and then evaluate the inference error and the induced delay.

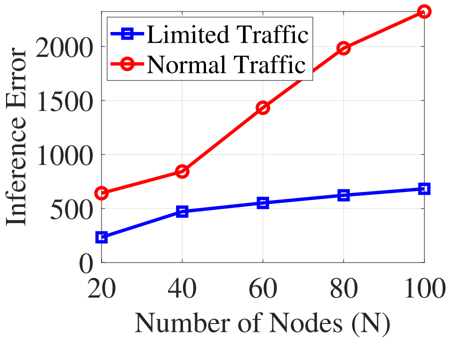

Figure 4: Genie bound of inference error with different number of nodes.

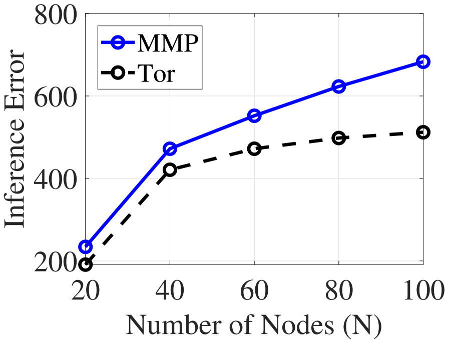

Figure 5: Genie bound of inference error between MMP and Tor.

5.1 Experimental Setups

5.1.1 Network Topology. The network is simulated using RGG with N nodes, and the number of nodes varies between 20 and 100, and randomly place them in a region $[0, \sqrt{N/\lambda}]$. We consider the node density $\lambda = 5$ indicating that the expected number of neighbors of each node is 5, and the communication range of each node as $r = 2$.

5.1.2 Parameter Setting. The theoretical results from Theorem **??** indicates that the inference error is associated with number of flows F, therefore, in our experiments, we consider two different traffic scenarios, i.e., 1) limited traffic where $F = \lfloor \sqrt{N} \rfloor$, 2) normal traffic where $F = N$. In the first scenario, only a few of nodes are involved in the communication process, whereas in the second scenario, almost all the nodes are active in at least one traffic flow. We consider the data rate of each flow x_i as a random variable subject to Gaussian distribution. The default mean and variance of x_i are set as $\mu = 10$ and $\sigma^2 = 2$ respectively.

5.1.3 Performance Metrics. In the experiments, we consider the worst-case that the inference algorithm is unknown, the genie bound is applied to gauge inference errors. Section **??** outlines the approach to derive this genie bound, while the delay is obtained by averaging the hop count across all network flows. For the purpose of comparative analysis with the results in [?], we also implement the Tor network as a typical framework for analyzing existing routing protocols.

5.2 Inference Error

We first evaluate the inference error of our proposed MMP protocol.

5.2.1 Varying Number of Nodes. Theorem **??** shows that the inference error increase quadratically if we increase the number of nodes. We compare the results between both traffic scenarios in Figure **??**. We can clear see that as we increase the number of nodes N, the difference of inference error between both traffic scenarios is enlarged. This is because the attacker need to guess more paths under the normal traffic making the attacker to have a correct guess on the routing matrix more difficult.

Figure **??** shows the difference between our proposed MMP and Tor. As can be seen, MMP provides a significantly better inference error than Tor. Tor requires the path to pass through the selected relays, thus decreasing the search space for selecting a path. The size of the path set of Tor for arbitrary flow is a subset of that of MMP. As a result, MMP has a higher inference error than Tor.

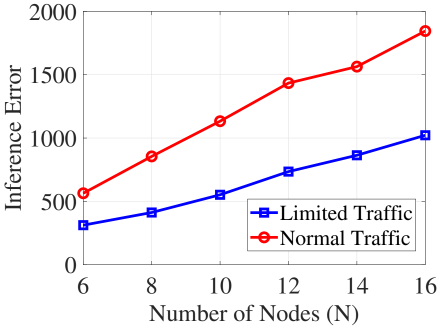

Figure 6: Genie bound of inference error between MMP and Tor.

5.2.2 Varying Mean Value. The inference error is also related to the mean value μ of the flow rate. Figure **??** shows the comparison results between limited traffic and normal traffic for different μs. In this figure, we can clear see that as we increase μ the inference rate of both flow scenarios increase. The inference error of normal traffic is universally better than the limited traffic because in the normal scenario, there are more flow that the attacker should guess.

5.3 Delay Evaluation

According to Theorem **??**, the delay is related to the number of nodes N. Figure **??** shows the evaluation results between delay and N. When the number of nodes increases, the delay for both traffic scenarios increase as well. In addition, there is no big difference between both traffic scenarios because the number of flows has little impact on the delay.

Figure **??** shows the difference between MMP and Tor in terms of the delay. MMP has a smaller delay when there are fewer than 100 nodes, and when the number is greater than 100, MMP has a larger delay. The reason behind this phenomenon is that when the number of nodes is not sufficient, MMP is more likely to choose

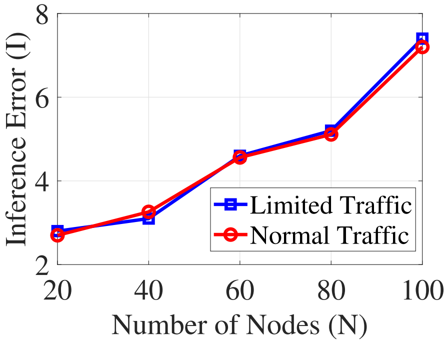

Figure 7: The delay of MMP for different number of nodes.

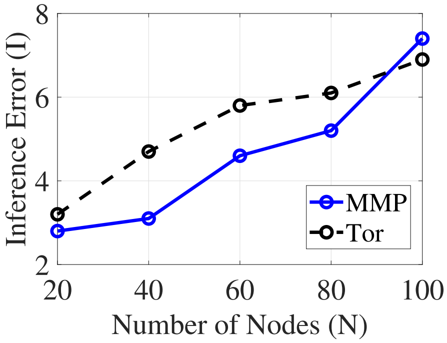

Figure 8: The delay of MMP for different number of nodes.

the path with a smaller delay. For Tor, however, the delay is at least 3, indicating three relays were selected.

6 RELATED WORK

Our work is related to the network inference and tomography, and the dynamic or random routing designs.

6.1 Network Inference and Tomography

Network tomography and inference have emerged as essential techniques in the field of network monitoring and analysis [? ?]. Network inference involves the estimation of flow information through eavesdropping on wireless link activities, which is widely feasible in wireless networks because of the broadcast nature of the wireless medium [? ? ? ?]. Many of existing works of network inference and tomography were designed for the optimizing the inference accuracy [? ? ? ? ? ? ?]. However, the applicability of network inference is still limited by some strong assumptions (e.g., known network topology, etc.). In light of the aforementioned problem, machine learning can be used in order to predict the underlying unknown parameters [? ? ? ?]. For example, the authors in [?] use deep neural networks to predict unmeasured network attributes and reconstruct network topology. In [?] leveraged machine learning to facilitate the network inference when only limited information about the network's topology is available.

From the security perspective, using network inference, the attacker can obtain internal information of a network without accessing to it. In terms of the attack, the authors in [? ?] proposed an data poisoning attack targeting on misleading the network operator to make a wrong decision. The authors in [? ?] analyzed the fundamental limit of a stealthy attacker in maximally degrading the performance of end-to-end communications without being localized. For the defense, the authors in [? ? ?] provided the proactive strategy to intentionally degree the inference performance. For example, in [?], authors provide a theoretical analysis on the relationship between the inference error and the artificial noise added on the measurements. Authors in [?], by integrating the machine learning, authors noticed that the network topology can be obfuscated to attackers. In our early work [?], the performance of many existing dynamic routing protocols was investigated and compared without proposing new protocols. However, we notice that most of existing protocols prioritize security objectives other than defending network inferences, thereby hindering resilience performance. Different from previous works, this paper focus on investigating the reason to cause the inference error and then designing a routing protocol that specifically addresses the resilience of the network inference attack on wireless networks

6.2 Dynamic Routing

Dynamic and random routing strategies have garnered significant attention in the wireless networks, offering promising solutions to the challenges posed by the ever-changing and unpredictable nature of wireless environments [? ?]. Dynamic routing adapts to network dynamics by selecting paths for data transmission based on real-time conditions such as signal strength[? ? ?], traffic load [?], channel interference[? ?], and topology changes [?]. For example, in [?], the authors make relay selection based on instantaneous Received Signal Strength Indicator (RSSI) and Link Quality Indicator (LQI) values to fit the wireless environment. When it comes to defending against network inference attacks, dynamic routing can help enhance security by introducing variability into network traffic patterns. Previous research in [? ?] has demonstrated the basic idea that random routing can make the inference process inaccurate, and provide the analysis results on existing dynamic routing protocols. However, whether such protocols can provide sufficient randomness against network inference attacks remain unclear. In stead, in this paper, we observed that the inference error is positively related to the probability of the mismatch between the flow template observed by the attackers, and the real template used in the network. Motivated by this, we propose a dynamic routing protocol, called Max-Mismatch-Probability (MMP), which seeks to maximize mismatch probability and increase the inference error.

7 CONCLUSION

Attacks employing network inference pose a significant threat to network security since they provide attackers with the opportunity to gain insight into sensitive flow information without gaining direct access to it. Using flow information, an attacker can launch powerful attacks against the wireless network. In wireless networks, dynamic routing can enhance security and privacy by increasing variability in traffic patterns and thereby increasing inference error.

In this paper, we propose a new dynamic routing protocol, called MMP, that maximizes inference error, thus effectively hiding flow information. We conduct a comprehensive theoretical analysis of the inference errors of MMP against inference attacks and the cost of the protocol, which is the delay. Using MMP, we have shown that the flow information can be concealed effectively.

ACKNOWLEDGEMENT

This work was supported in part by NSF under grants SaTC-2321271 and CNS-2312138.

REFERENCES

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

[2] Novella Bartolini, Ting He, Viviana Arrigoni, Annalisa Massini, Federico Trombetti, and Hana Khamfroush. On fundamental bounds on failure identifiability by boolean network tomography. *IEEE/ACM Transactions on Networking*, 28(2):588–601, 2020.

[3] Ilker Bekmezci, Ozgur Koray Sahingoz, and Şamil Temel. Flying ad-hoc networks (fanets): A survey. *Ad Hoc Networks*, 11(3):1254–1270, 2013.

[4] Sanjit Biswas and Robert Morris. Opportunistic routing in multi-hop wireless networks. *ACM SIGCOMM Computer Communication Review*, 34(1):69–74, 2004.

[5] Emmanuel J Candès, Justin Romberg, and Terence Tao. Robust uncertainty principles: Exact signal reconstruction from highly incomplete frequency information. *IEEE Trans. Inf. Theory*, 52, 2006.

[6] Rui Castro, Mark Coates, Gang Liang, Robert Nowak, and Bin Yu. Network tomography: Recent developments. 2004.

[7] Szymon Chachulski, Michael Jennings, Sachin Katti, and Dina Katabi. Trading structure for randomness in wireless opportunistic routing. *ACM SIGCOMM Computer Communication Review*, 37(4):169–180, 2007.

[8] Nessrine Chakchouk. A survey on opportunistic routing in wireless communication networks. *IEEE Communications Surveys & Tutorials*, 17(4):2214–2241, 2015.

[9] Cho-Chun Chiu and Ting He. Stealthy dgos attack against network tomography: The role of active measurements. *IEEE Transactions on Network Science and Engineering*, 8(2):1745–1758, 2021.

[10] Cho-Chun Chiu and Ting He. Stealthy dgos attack: Degrading of service under the watch of network tomography. *IEEE/ACM Transactions on Networking*, 29(3):1294–1307, 2021.

[11] Xiangrui Fan, Wenlong Cai, and Jinyong Lin. A survey of routing protocols for highly dynamic mobile ad hoc networks. In *2017 IEEE 17th International Conference on Communication Technology (ICCT)*, pages 1412–1417. IEEE, 2017.

[12] XiaoBo Fan and Xingming Li. Network tomography via sparse bayesian learning. *IEEE Communications Letters*, 21(4):781–784, 2017.

[13] Mohammad Hamed Firooz and Sumit Roy. Link delay estimation via expander graphs. *IEEE Trans. Commun.*, 62:170–180, 2014.

[14] Manjesh K Hanawal, Diep N Nguyen, and Marwan Krunz. Jamming attack on in-band full-duplex communications: Detection and countermeasures. In *IEEE INFOCOM*, 2016.

[15] Ting He. Distributed link anomaly detection via partial network tomography. 2018.

[16] Tao Hou, Zhe Qu, Tao Wang, Zhuo Lu, and Yao Liu. Proto: Proactive topology obfuscation against adversarial network topology inference. In *IEEE INFOCOM 2020-IEEE Conference on Computer Communications*, pages 1598–1607. IEEE, 2020.

[17] Yiyi Huang, Nick Feamster, and Renata Teixeira. Practical issues with using network tomography for fault diagnosis. *ACM SIGCOMM Computer Communication Review*, 38(5):53–58, 2008.

[18] Amani Ibraheem, Zhengguo Sheng, George Parisis, and Daxin Tian. Neural network based partial tomography for in-vehicle network monitoring. In *2021 IEEE International Conference on Communications Workshops (ICC Workshops)*, pages 1–6. IEEE, 2021.

[19] Amani Ibraheem, Zhengguo Sheng, George Parisis, Jianshan Zhou, and Daxin Tian. Internal network monitoring with dnn and network tomography for in-vehicle networks. In *2022 IEEE International Conference on Unmanned Systems (ICUS)*, pages 928–933. IEEE, 2022.

[20] Grigorios Kakkavas, Despoina Gkatzioura, Vasileios Karyotis, and Symeon Papavassiliou. A review of advanced algebraic approaches enabling network tomography for future network infrastructures. *Future Internet*, 12(2):20, 2020.

[21] Hiroyuki Kasai, Wolfgang Kellerer, and Martin Kleinsteuber. Network volume anomaly detection and identification in large-scale networks based on online time-structured traffic tensor tracking. *IEEE Trans. Netw. Service Manag.*, 13, 2016.

[22] Mohammad Shoeb Saeed Khan. *Network tomography application in mobile ad-hoc networks.* University of Louisville, 2013.

[23] Demeke Shumeye Lakew, Umar Sa'ad, Nhu-Ngoc Dao, Woongsoo Na, and Sungrae Cho. Routing in flying ad hoc networks: A comprehensive survey. *IEEE Communications Surveys & Tutorials*, 22(2):1071–1120, 2020.

[24] Fengyin Li, Pei Ren, Guoyu Yang, Yuhong Sun, Yilei Wang, Yanli Wang, Siyuan Li, and Huiyu Zhou. An efficient anonymous communication scheme to protect the privacy of the source node location in the internet of things. *Security and Communication Networks*, 2021:1–16, 2021.

[25] Yimei Li and Yao Liang. Compressed sensing in multi-hop large-scale wireless sensor networks based on routing topology tomography. *IEEE Access*, 6:27637–27650, 2018.

[26] Yongjun Li, Wandong Cai, Guangli Tian, and Wei Wang. Loss tomography in wireless sensor network using gibbs sampling. In *Wireless Sensor Networks: 4th European Conference, EWSN 2007, Delft, The Netherlands, January 29-31, 2007. Proceedings 4*, pages 150–162. Springer, 2007.

[27] Yunzhong Liu, Rui Zhang, Jing Shi, and Yanchao Zhang. Traffic inference in anonymous manets. In *IEEE SECON*, 2010.

[28] Zhuo Lu and Cliff Wang. Network anti-inference: A fundamental perspective on proactive strategies to counter flow inference. In *IEEE INFOCOM*, 2015.

[29] Zhuo Lu and Cliff Wang. Enabling network anti-inference via proactive strategies: A fundamental perspective. *IEEE/ACM Transactions on Networking*, 25(1):43–55, 2016.

[30] Liang Ma, Ting He, Kin K Leung, Ananthram Swami, and Don Towsley. Identifiability of link metrics based on end-to-end path measurements. In *ACM IMC*, 2013.

[31] Liang Ma, Ting He, Kin K Leung, Don Towsley, and Ananthram Swami. Efficient identification of additive link metrics via network tomography. In *2013 IEEE 33rd International Conference on Distributed Computing Systems*, pages 581–590. IEEE, 2013.

[32] Liang Ma, Ziyao Zhang, and Mudhakar Srivatsa. Neural network tomography. *arXiv preprint arXiv:2001.02942*, 2020.

[33] Morteza Mardani and Georgios B Giannakis. Estimating traffic and anomaly maps via network tomography. *IEEE/ACM Trans. Netw.*, 24, 2016.

[34] Takahiro Matsuda, Masaaki Nagahara, and Kazunori Hayashi. Link quality classifier with compressed sensing based on\ell_1-\ell_2 optimization. *IEEE Communications Letters*, 15(10):1117–1119, 2011.

[35] Lu Mei-Hsuan, Steenkiste Peter, and Chen Tsuhan. Design, implementation and evaluation of an efficient opportunistic retransmission protocol. *Proc. Of IEEE MobiCom, Beijing, China*, 2009.

[36] Mathew Penrose. *Random geometric graphs*, volume 5. OUP Oxford, 2003.

[37] Ippokratis Sartzetakis and Emmanouel Varvarigos. Machine learning network tomography with partial topology knowledge and dynamic routing. In *GLOBECOM 2022-2022 IEEE Global Communications Conference*, pages 4922–4927. IEEE, 2022.

[38] Anirvan M Sengupta and Partha P Mitra. Distributions of singular values for some random matrices. *Physical Review E*, 60, 1999.

[39] Rahul C Shah, Sven Wietholter, and Adam Wolisz. Modeling and analysis of opportunistic routing in low traffic scenarios. In *Third International Symposium on Modeling and Optimization in Mobile, Ad Hoc, and Wireless Networks (WiOpt'05)*, pages 294–304. IEEE, 2005.

[40] Rajinder Singh and Satish Kumar. A comparative study of various wireless network monitoring tools. In *2018 First International Conference on Secure Cyber Computing and Communication (ICSCCC)*, pages 379–384. IEEE, 2018.

[41] Paul Syverson, R Dingledine, and N Mathewson. Tor: The secondgeneration onion router. In *USENIX Security*, 2004.

[42] Yehuda Vardi. Network tomography: Estimating source-destination traffic intensities from link data. *Journal of the American statistical association*, 91(433):365–377, 1996.

[43] Wei Wang, Huiran Wang, Beizhan Wang, Yaping Wang, and Jiajun Wang. Energy-aware and self-adaptive anomaly detection scheme based on network tomography in mobile ad hoc networks. *Information Sciences*, 220:580–602, 2013.

[44] Zehua Wang, Yuanzhu Chen, and Cheng Li. Corman: A novel cooperative opportunistic routing scheme in mobile ad hoc networks. *IEEE journal on selected areas in communications*, 30(2):289–296, 2012.

[45] Chung-Kai Yu, Kwang-Cheng Chen, and Shin-Ming Cheng. Cognitive radio network tomography. *IEEE Trans. Veh. Technol.*, 59, 2010.

[46] Zhenghao Zhang and Avishek Mukherjee. Friendly channel-oblivious jamming with error amplification for wireless networks. In *IEEE INFOCOM*, 2016.

[47] Zhiyong Zhang, Ovidiu Mara, and Katerina Argyraki. Network neutrality inference. In *ACM SIGCOMM*, 2014.

[48] Jerry Zhao, Ramesh Govindan, and Deborah Estrin. Sensor network tomography: Monitoring wireless sensor networks. *ACM SIGCOMM Computer Communication Review*, 32(1):64–64, 2002.

[49] Shangqing Zhao, Zhuo Lu, and Cliff Wang. When seeing isn't believing: On feasibility and detectability of scapegoating in network tomography. In *IEEE ICDCS*, 2017.

[50] Shangqing Zhao, Zhuo Lu, and Cliff Wang. How can randomized routing protocols hide flow information in wireless networks? *IEEE Transactions on Wireless Communications*, 19(11):7224–7236, 2020.

[51] Shangqing Zhao, Zhuo Lu, and Cliff Wang. Measurement integrity attacks against network tomography: Feasibility and defense. *IEEE Transactions on Dependable and Secure Computing*, 18:2617–2630, Nov. 2021.

[52] Zhonghua Zhao, Wei Huangfu, and Linmin Sun. Nssn: A network monitoring and packet sniffing tool for wireless sensor networks. In *2012 8th International Wireless Communications and Mobile Computing Conference (IWCMC)*, pages 537–542. IEEE, 2012.

[53] Zhongliang Zhao, Denis Rosário, Torsten Braun, Eduardo Cerqueira, Hongli Xu, and Liusheng Huang. Topology and link quality-aware geographical opportunistic routing in wireless ad-hoc networks. In *2013 9th international wireless communications and mobile computing conference (IWCMC)*, pages 1522–1527. IEEE, 2013.

[54] Lan Zhuo, Yutong Li, Jun Deng, and Hao Wang. An anonymous communication method for wireless sensor networks based on bilinear pairings. In *2020 IEEE 2nd International Conference on Civil Aviation Safety and Information Technology (ICCASIT*, pages 517–525. IEEE, 2020.