

Parsimonious Tomography: Optimizing Cost-Identifiability Trade-off for Probing-based Network Monitoring

Diman Zad Tootaghaj, Ting He, Thomas La Porta
The Pennsylvania State University
`{dxz149, tzh58, tlp}@cse.psu.edu`

ABSTRACT

Network tomography using end-to-end probes provides a powerful tool for monitoring the performance of internal network elements. However, active probing can generate tremendous traffic, which degrades the overall network performance. Meanwhile, not all the probing paths contain useful information for identifying the link metrics of interest. This observation motivates us to study the optimal selection of monitoring paths to balance identifiability and probing cost. Assuming additive link metrics (e.g., delays), we consider four closely-related optimization problems: 1) *Max-IL-Cost* that maximizes the number of identifiable links under a probing budget, 2) *Max-Rank-Cost* that maximizes the rank of selected paths under a probing budget, 3) *Min-Cost-IL* that minimizes the probing cost while preserving identifiability, and 4) *Min-Cost-Rank* that minimizes the probing cost while preserving rank. While (1) and (3) are hard to solve, (2) and (4) are easy to solve, and the solutions give a good approximation for (1) and (3). Specifically, we provide an optimal algorithm for (4) and a $(1 - 1/e)$ -approximation algorithm for (2). We prove that the solution for (4) provides tight upper/lower bounds on the minimum cost of (3), and the solution for (2) provides upper/lower bounds on the maximum identifiability of (1). Our evaluations on real topologies show that solutions to the rank-based optimization (2, 4) have superior performance in terms of the objectives of the identifiability-based optimization (1, 3), and our solutions can reduce the total probing cost by an order of magnitude while achieving the same monitoring performance.

Categories and Subject Descriptors

C.2.3 [Computer-communication Networks]: Network Operations—*Network Monitoring*; G.2.2 [Discrete Mathematics]: Graph Theory—*Network problems*

Keywords

Network Tomography; Identifiability; Path Selection

1. INTRODUCTION

Today's Internet traffic is massive, heterogeneous, and distributed, and continues to grow in these dimensions. Therefore, unlike small-scale networks, provisioning the

desired services under an acceptable quality of service (QoS) for the ever-growing traffic sizes is extremely challenging and depends on continuous monitoring of the performance of individual links. Network monitoring provides the internal network state that is crucial for many network management functions such as traffic engineering, anomaly detection, and service provisioning. In cases where the important performance metrics are not directly observable (e.g., due to lack of access), network tomography provides a solution that infers these metrics from end-to-end probes [1–4]. Compared to other monitoring techniques such as SNMP polling, *ping*, or *traceroute*, end-to-end probes does not need any special support from the routers [5–9] and is therefore a reliable tool for monitoring the Internet.

However, despite the considerable amount of research on estimating the individual link's performance metrics using given end-to-end measurements, the selection of which paths to probe, either to minimize probing cost or to satisfy a given bound (i.e., budget) on the probing cost, has not been thoroughly studied in prior works. Probing all possible paths between each pair of monitors can produce a tremendous amount of traffic in the network. Meanwhile, many paths contain redundant information due to shared links. In this paper, we show that by carefully selecting the probing paths, we can significantly reduce the amount of probing traffic while achieving the same monitoring performance.

To this end, we consider the following closely-related problems under the assumption of additive performance metrics (e.g. delays): 1) the ***Max-IL-Cost*** problem that maximizes the number of identifiable links under a limited probing budget, 2) the ***Max-Rank-Cost*** problem that maximizes the rank of probing paths under a probing budget, 3) the ***Min-Cost-IL*** problem that minimizes the probing cost while identifying all the identifiable links, 4) the ***Min-Cost-Rank*** problem that minimizes the probing cost while preserving the rank. Problems (1) and (3) are considered because they address, from different perspectives, the optimal trade-off between monitoring performance (measured by identifiability) and probing cost. Problems (2) and (4) are con-

sidered because they possess desirable properties that allow efficient computation while providing good approximations to (1) and (3).

Specifically, we make the following contributions:

1. Based on an existing algorithm that computes all the minimal sets of paths to identify each identifiable link, we convert (1) and (3) to problems similar to the max-coverage problem [10] and the set-cover problem [11], respectively. The conversion allows us to apply the greedy heuristic to these problems. We also propose an iterative branch-and-bound algorithm that treats our problems as integer linear programs (ILPs), and decomposes each problem into smaller meaningful sub-problems to exploit parallelism on a multi-core machine. Using our iterative branch-and-bound algorithm, we can configure the trade-off between the execution time and the optimality gap of the solution.
2. Using techniques from matroid optimization, we give polynomial-time solutions to (2) and (4) with guaranteed performance. The proposed solution for (4) is provably optimal, and the solution for (2) achieves a $(1 - 1/e)$ -approximation.
3. We show that the solution for (4) provides tight upper/lower bounds for (3), and the solution for (2) provides upper/lower bounds for (1).
4. Our evaluations on real topologies show that in terms of the objectives of (1) and (3), the solutions proposed for (2) and (4) perform very close to the optimal and even outperform the solutions designed for (1) and (3). Compared to the baseline of probing all the candidate probing paths, our solutions can reduce the probing cost by an order of magnitude while achieving the same monitoring performance.

The remainder of this paper is organized as follows. Section 2 discusses the background and motivation behind this work. In section 3, we formulate the four optimization problems. Section 4 contains our algorithms and their performance analysis. Section 5 shows our evaluation methodology and results. Finally, Section 6 concludes the paper.

2. BACKGROUND AND MOTIVATION

2.1 Background

The problem of designing the monitoring system to optimize the trade-off between cost and monitoring performance is a long-standing hard problem. If monitors cannot control the routing of probes, the problem is to place the minimum number of monitors (beacons) to identify all the links, which is proved to be NP-hard

[12,13]. If monitors can control the probing paths (e.g., via source routing or software-defined networking), the problem is to both place the minimum number of monitors and construct the minimum number of probing paths to identify all the links, which is polynomial-time solvable [14–16]. In contrast to [14–16], we assume that routes cannot be controlled, as is usually the case in IP networks; in contrast to [12,13] that focus on the offline cost for deploying monitors, we focus on the online cost for sending probes (i.e., the probing cost).

In the context of overlay networks, Chen et al. [17] show that monitoring a set of $O(n \log(n))$ paths is sufficient for monitoring an overlay network of n hosts, by selecting a set of paths that gives a basis of all the paths between the hosts. Li et al. propose a polynomial-time path selection algorithm that minimizes the total cost of selected paths to cover all the links [18]. These approaches differ from ours in that they focus on end-to-end performance, e.g. loss rates for all end-to-end paths, while we focus on identifying the performance of each individual link.

Zheng et al. [19] introduced a problem similar to our third optimization (*Min-Cost-IL*). They study the problem of selecting the minimum number of probing paths that can uniquely identify all the identifiable links and cover all the unidentifiable links. Our formulation differs from theirs in that we allow general probing costs for the paths, and do not require coverage of all the links. These differences allow us to model paths with heterogeneous probing costs and further reduce the total cost without losing identifiability. More importantly, the solution in [19] requires the calculation of all the irreducible path sets to identify each of the identifiable links, which has an exponential complexity. In contrast, we show that using rank as a proxy of identifiability gives an efficient solution that provides tight upper/lower bounds on the optimal solution (Theorem 3).

2.2 Motivation

We use an example to illustrate the cost saving that can be achieved by a careful selection of probing paths. Suppose that the cost of probing a path is equal to the total number of links on this path, which represents the amount of traffic that each probe on this path will generate. We consider three networks from the Internet Topology Zoo [20,21], randomly select a subset of nodes in each network as monitors, and compute the shortest paths between each pair of monitors as the candidate probing paths. As shown in Table 1, probing all these paths generates a large number of transmissions and incurs a high cost (total cost). In contrast, using a selected subset of paths that preserve the rank (computed by Algorithm 2), we can obtain the same information at a much lower cost (cost of selected paths). As is shown, using the selected paths reduces the probing cost by a

Table 1: Cost Reduction Using Selected Probing Paths

network name	#monitors	#paths	total cost	#selected paths	cost of selected paths
Abilene	11	55	244	12	46
BellCanada	20	190	4467	31	284
CAIDA	34	528	25553	56	1606

factor of 5.3–16 in this example. The large gap between the total probing cost and the probing cost of the given paths motivates the study of the path selection problem.

3. PROBLEM FORMULATION

In this section, we describe our network model, performance measures and optimization problems.

3.1 Network Model

Given an undirected graph $G(V, L)$, where V represents the network nodes and L is the set of communicating links connecting the nodes, and a set of nodes $M \subseteq V$ employed as monitors, the set P of routing paths between all pairs of monitors specifies the set of candidate probing paths that we can select from. Each link j in L is associated with an additive metric x_j (e.g., link delay). In our model, we assume IP packets from a source node s to a destination node t are being forwarded using a pre-determined routing algorithm. Our formulation and solutions support arbitrary routing algorithms, and the specific algorithm used for evaluation will be specified later (see Section 5). We denote a routing path r in G with a list of edges $r = \{e_1, \dots, e_n\}$ and denote with k_r the cost of path r . Probes on each path r in P give the end-to-end metric of this path y_r . Given a set P of all possible probing paths (e.g., routing paths between all the monitors), let A be the routing matrix of size $|P| \times |L|$, such that if path $r \in P$ contains link j , then $A[r, j] = 1$ and $A[r, j] = 0$ otherwise. We can write a linear system of equations relating the link's additive metric (e.g. delay) to path metrics as $Ax = y$. The objective of network tomography is to infer x from A and y . Table 2 summarizes the notation used in our formulation.

3.2 Measures of Monitoring Performance

The linear system of equations (introduced in Section 3.1) may not be invertible as the routing matrix A may not have a full column rank. To quantify the extent to which this system can be solved, we introduce two measures: **identifiability** and **rank**. The rank of P is calculated by the rank of the routing matrix A , denoted by $\text{rank}(A)$, which is the cardinality of the largest set of probing paths, such that each path in the set contains “new information” about the links (every other path is a linear combination of paths in the set and thus does not provide new information).

A link j in L is identifiable using a set P of prob-

Table 2: Notation used in our formulations.

Notation	Explanation
$G(V, L)$	an undirected graph where V represents the set of nodes and L is the set of links
L_I	the set of identifiable links using all possible paths P
M	set of nodes where the monitors are located
$I(P_R)$	set of all identifiable links using paths in P_R
K	limit on the probing cost
$P_l := \{P_{l_i} : i = 1, \dots, S_l\}$	set of all minimal solutions to link $l \in L$
X_l	decision to select an identifiable link l (if $X_l = 1$) or not (if $X_l = 0$)
Y_r	decision to select a path r (if $Y_r = 1$) or not (if $Y_r = 0$)
Z_s	decision to select a minimal solution s (if $Z_s = 1$) or not (if $Z_s = 0$)
P	the total set of paths using all of the monitoring nodes M
P_R	a subset of all possible probing paths with indices in R
$r_{u,v}$	given a source node u and a destination node v and a pre-defined routing algorithm, $r_{u,v}$ gives the routing path from u to v
A_R	a routing matrix of size $ R \times L $, such that if path $r \in R$ contains link j , then $A_R[r, j] = 1$ and $A_R[r, j] = 0$ otherwise
$\text{rank}(A_R)$	the rank of routing matrix A_R
k_r	probing cost of path r
$c(P_R) = \sum_{r \in P_R} k_r$	total cost of a set of probing paths $P_R \subseteq P$
K	limit on probing cost

ing paths if its metric can be uniquely determined from the metrics of the paths in P . The identifiability of a network under probing paths P is the number of links that are identifiable using P . Let $N = \text{Null}(A)$ denote the null space of A , i.e. for any vector $n \in \text{Null}(A)$, $A \cdot n = 0$. The next lemma specifies how to compute the set of identifiable links given A .

LEMMA 1. [17] *For an arbitrary routing matrix A , let N represent the null space of A . Link $l_i \in L$ is identifiable, if and only if $\forall n \in N$ we have $n_i = 0$.*

Therefore, to find the set of identifiable links, $L_I \in L$, we first compute the null space of A and find all indices with zero values in the null space. The identifiability achieved by probing P is then the cardinality of L_I .

3.2.1 Relationship between Identifiability and Rank

While identifiability is a more accurate measure of the usefulness of the paths for network tomography, rank is easier to optimize as is shown later (see Section 4.2). Below, we establish the relationship between the two measures.

Let a set P of routing paths $\{r_1, \dots, r_n\}$ be given. Corresponding to any subset $P_R \subseteq P$ of these elements, let $\text{rank}(A_R)$ be the rank of the routing matrix corresponding to the selected paths in P_R . We define L_1 to be any subset of identifiable links ($L_1 \subseteq L_I$) and pro-

vide the necessary and sufficient condition for a subset of paths to identify all identifiable links.

THEOREM 1. Given a subset of paths $P_R \subseteq P$ and a subset of links $L' \subseteq L$, let A_R be a sub-matrix of A generated by selecting all the rows corresponding to paths in P_R , and $A_{R,L'}$ be a sub-matrix of A_R generated by selecting all the columns corresponding to links in L' . A subset of paths $P_R \subseteq P$ can identify a subset of links $L_1 \subseteq L$ if and only if

$$\text{rank}(A_R) = |L_1| + \text{rank}(A_{R,L \setminus L_1}), \quad (1)$$

PROOF. In Section 8 (APPENDIX). \square

An illustrative example: Figure 1 shows an example of a network with 5 links and four candidate monitors $M = \{m_1, \dots, m_4\}$. Using all possible paths between candidate monitors we have the following routing matrix.

$$A = \begin{pmatrix} l_1 & l_2 & l_3 & l_4 & l_5 \\ \hline 1 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 1 & 1 \\ 0 & 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 & 1 \end{pmatrix} : r_{m_1,m_2} \\ r_{m_1,m_3} \\ r_{m_1,m_4} \\ r_{m_2,m_3} \\ r_{m_2,m_4} \\ r_{m_3,m_4}$$

$$A_{*,L_1} \quad A_{*,L \setminus L_1}$$

The rank of this matrix is 4 while the null space shows only 3 identifiable links l_1, l_2, l_3 . If we only probe paths in $R = \{r_{m_1,m_2}, r_{m_1,m_3}, r_{m_2,m_3}\}$, the corresponding routing matrix A_R satisfies Theorem 1.

$$A_R = \begin{pmatrix} l_1 & l_2 & l_3 & l_4 & l_5 \\ \hline 1 & 1 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 \end{pmatrix} : r_{m_1,m_2} \\ r_{m_1,m_3} \\ r_{m_2,m_3}$$

$$A_{R,L_1} \quad A_{R,L \setminus L_1}$$

Meanwhile, it is also clear that probing these paths suffices to identify l_1, l_2 and l_3 . We can solve the identifiable links using Gaussian elimination, where the reduced row echelon form ($rref(A)$) is

$$\left(\begin{array}{ccc|cc} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ \hline 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{array} \right) \quad \begin{array}{l} l_1 : (r_{m_1,m_2} + r_{m_1,m_3} - r_{m_2,m_3})/2 \\ l_2 : (r_{m_1,m_2} + r_{m_2,m_3} - r_{m_1,m_3})/2 \\ l_3 : (r_{m_1,m_3} + r_{m_2,m_3} - r_{m_1,m_2})/2 \\ l_4 + l_5 \\ 0 \\ 0 \end{array}$$

As shown, the reduced row echelon form contains an identity matrix for columns corresponding to identifiable links and by choosing $\{r_{m_1,m_2}, r_{m_1,m_3}, r_{m_2,m_3}\}$, the conditions of theorem 1 are satisfied. Further, the total probing cost reduces from 15 to 6.

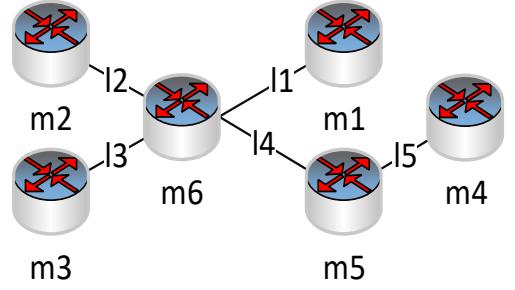


Figure 1: A simple network example with 5 links and 4 monitors $\{m_1, \dots, m_4\}$. Candidate paths: $r_{m_1,m_2}, r_{m_1,m_3}, r_{m_1,m_4}, r_{m_2,m_3}, r_{m_2,m_4}, r_{m_3,m_4}$.

3.3 Optimization Problems

3.3.1 Max-IL-Cost problem

Let $I(P_R)$ be the set of identifiable links using paths in P_R and $|I(P_R)|$ be the number of identifiable links using paths in P_R . The constrained path selection optimization problem aims at maximizing the number of identifiable links (*Max-IL-Cost*) with a limited probing cost K , which can be formulated as follows:

$$\text{Maximize } |I(P_R)| \quad (2a)$$

$$\text{subject to } \sum_{r \in P_R} k_r \leq K, \quad (2b)$$

$$P_R \subseteq P, \quad (2c)$$

where k_r is the probing cost of path r . As a concrete example, we can define the probing cost of each path to be its total number of hops. Then the total probing cost represents the total number of transmissions generated by probing the selected set of paths.

ILP formulation for Max-IL-Cost: To better understand properties of Max-IL-Cost, we re-write it as an integer linear programming (*ILP*). The basis of our formulation is the notion of *minimal solutions* (simply called *solutions* in [19]). Each minimal solution to link $l \in L$ is a subset of paths $P' \subseteq P$ such that (i) P' can identify l , but (ii) no proper subset of P' can identify l . As an example, consider the network in Figure 1. Consider the following two sets of paths $P_1 = \{r_{m_1,m_2}, r_{m_1,m_3}, r_{m_2,m_3}\}$ and $P_2 = \{r_{m_2,m_3}, r_{m_2,m_4}, r_{m_3,m_4}\}$, which are both minimal solutions to link l_2 .

We can compute all the minimal solutions for each link l by first finding a solution to l and then use a linear replacement method to generate other solutions, as described in [19]. Let \mathcal{P}_l be the set of all the minimal solutions to link $l \in L$ ($\mathcal{P}_l = \emptyset$ if l is not identifiable). Then $\mathcal{P} = \bigcup_{l \in L} \mathcal{P}_l := \{P_s\}_{s \in S}$ is the collection of all

the minimal solutions for the identifiable links. For ease of presentation, we index the solutions in \mathcal{P} and denote by S the set of solution indices.

Based on the minimal solutions, we can write the Max-IL-Cost problem as the following ILP:

$$\text{Maximize}_{X_l, Y_r, Z_s} \sum_{l \in L} X_l \quad (3a)$$

$$\text{subject to } X_l \leq \sum_{s: l \in I(P_s)} Z_s, \quad \forall l \in L, \quad (3b)$$

$$\sum_{r \in P} Y_r \cdot k_r \leq K, \quad (3c)$$

$$Z_s \leq Y_r, \quad \forall s \in S, r \in P_s, \quad (3d)$$

$$X_l, Y_r, Z_s \in \{0, 1\}, \quad \forall l \in L, r \in P, s \in S. \quad (3e)$$

Here the binary variables X_l , Y_r and Z_s respectively represent the decision to select an identifiable link (if $X_l = 1$), a probing path (if $Y_r = 1$), and a minimal solution (if $Z_s = 1$).

First, we show that given solutions to Y_r 's, the ILP is easy to solve.

LEMMA 2. *The ILP optimization problem can be relaxed over the integer variables X_l and Z_s and still gives an optimal integer solution.*

PROOF. In Section 8 (APPENDIX). \square

Remark: While the problem can be relaxed over X_l and Z_s , finding all minimal solutions has an exponential complexity. Furthermore, similar to [19], optimizing Y_r 's is hard to solve. Therefore, we use the rank function as a proxy to identifiability in Section 3.3.2 and show the upper/lower bounds for identifiability.

3.3.2 Max-Rank-Cost Problem

Creating all the minimal solutions in Max-IL-Cost has an exponential complexity which limits the scale of applicability to small networks. Therefore, we replace the identifiability measure in this problem by rank. The resulting optimization derived from Max-IL-Cost, referred to as *Max-Rank-Cost*, is formulated as follows:

$$\text{Maximize } \text{rank}(A_R) \quad (4a)$$

$$\text{subject to } \sum_{r \in P_R} k_r \leq K, \quad (4b)$$

$$P_R \subseteq P. \quad (4c)$$

The rank function has an important property that makes its maximization easy to solve. To this end, we introduce the following definition.

Submodularity Let P be a finite ground set. A set function $f : 2^P \rightarrow \mathbb{R}$ is submodular if for all sets $P_a, P_b \subseteq P$, we have

$$f(P_a \cup P_b) + f(P_a \cap P_b) \leq f(P_a) + f(P_b). \quad (5)$$

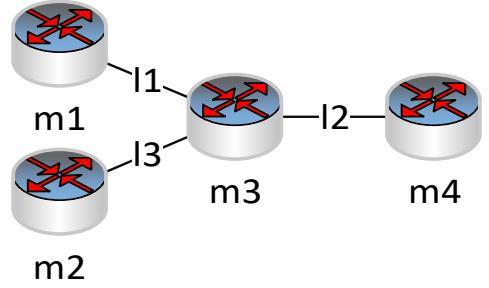


Figure 2: An example that shows identifiability is not a submodular or supermodular function.

Intuitively, f is a *submodular* function if it has the property of *diminishing return*, i.e., the marginal gain of adding an element e to a set P_a is at least as high as the marginal gain of adding e to any superset of P_a .

The significance of this property is that if $f(P)$ is monotone (i.e., increasing as we add elements to P) and submodular, then there is a generic greedy algorithm in [10] for maximizing $f(P)$ subject to a budget on P , which is within a $(1 - 1/e)$ -factor of the optimal. It is known that the rank function is submodular.

LEMMA 3. [22] *The rank function is monotone and submodular.*

However, the number of identifiable links $|I(P)|$ is not submodular. To see this, consider the example in Figure 2, which shows a network with 4 monitoring nodes (m_1, m_2, m_3, m_4). Consider the following path sets: $P_a = \{l_2\}$, and $P_b = \{(l_1, l_2), (l_3, l_2)\}$, where (l_i, l_j) denotes a 2-hop path traversing links l_i and l_j . Then it is easy to see that $I(P_a) = \{l_2\}$, $I(P_b) = \emptyset$, $I(P_a \cup P_b) = \{l_1, l_2, l_3\}$, and $I(P_a \cap P_b) = \emptyset$. Thus,

$$|I(P_a \cup P_b)| + |I(P_a \cap P_b)| > |I(P_a)| + |I(P_b)|,$$

violating submodularity.

3.3.3 Min-Cost-IL Problem

The problem of preserving identifiability using minimum probing cost is the dual of Max-IL-Cost problem. As a special case, Zheng et al. [19] considered the same problem when $k_r = k$ (i.e. all the paths have an identical probing cost). They show that even the special case is NP-hard by giving a reduction from set cover problem. They proposed a heuristic-based approach to cover all links by enumerating all possible combination of equations/paths that can cover each identifiable link. The constructed bipartite graph is then used to select the minimum number of probing paths that can cover all links where set cover is a special case of the problem. They assume each probing path has the

same cost, while Min-Cost-IL allows non-uniform, heterogeneous costs. Furthermore, while [19] also requires the coverage of non-identifiable links, our proposed algorithm only selects minimal sets that identify identifiable links. The constrained path selection optimization problem to minimize the probing cost to identify all identifiable links (*Min-Cost-IL*) is formulated as follows:

$$\text{Minimize} \quad \sum_{r \in P_R} k_r \quad (6a)$$

$$\text{subject to} \quad |I(P_R)| = |I(P)|, \quad (6b)$$

$$P_R \subseteq P. \quad (6c)$$

ILP formulation for Min-Cost-IL: Similar to (3), we re-write Min-Cost-IL as an integer linear programming (*ILP*) as follows:

$$\text{Minimize}_{Y_r, Z_s} \quad \sum_{r \in P} Y_r k_r \quad (7a)$$

$$\text{subject to} \quad 1 \leq \sum_{s: l \in I(P_s)} Z_s, \quad l \in I(P_s), \quad (7b)$$

$$Z_s \leq Y_r, \quad \forall s \in S, r \in P_s, \quad (7c)$$

$$Y_r, Z_s \in \{0, 1\}, \quad \forall r \in P, s \in S. \quad (7d)$$

The optimization minimizes the total cost of selected paths. The first constraint indicates that at least one of the minimal solutions for each identifiable link should be selected. The second constraint indicates that if a minimal solution is selected, all paths in the minimal set should also be selected.

3.3.4 Min-Cost-Rank Problem

Similar to section 3.3.2, we define the *Min-Cost-Rank* problem as minimizing the probing cost (total hop-count) subject to preserving rank. Let $P = \{r_1, \dots, r_{|M|(|M|-1)/2}\}$ be the total set of paths using all monitors M and let $P_R \subseteq P$ be a subset of selected paths. We define the routing matrix A_R of size $|R| \times |L|$ to be a matrix consisting of 0 and 1s, such that if $r \in P_R$ contains link j then $A_R[r, j] = 1$ and $A_R[r, j] = 0$ otherwise. We aim to select a subset of paths, $P_R \subseteq P$ such that the rank of both matrices be the same.

$$\text{Minimize} \quad \sum_{r=1}^{|M|(|M|-1)/2} k_r Y_r \quad (8a)$$

$$\text{subject to} \quad \text{rank}(A_R) = \text{rank}(A) \quad (8b)$$

$$Y_r \in \{0, 1\} \quad (8c)$$

Where the binary variable Y_r represent the decision to select a probing path r in A_R and k_r is the probing cost of path r .

4. PATH SELECTION ALGORITHMS

In this section, we give different algorithms for the four optimization problems. We propose a greedy heuris-

tic and an iterative branch-and-bound algorithm for the *Max-Cost-IL* and the *Min-Cost-IL* problems. We also show a greedy algorithm that is optimal for *Min-Cost-Rank* and a modified greedy algorithm that achieves a $(1 - 1/e)$ -approximation for *Max-Rank-Cost*.

4.1 Algorithms for Identifiability Optimization

4.1.1 Greedy-Max-IL-Cost and Greedy-Min-Cost-IL

We explain how to select a given set of paths using a set of feasible monitors and a pre-defined routing algorithm. To compare with the existing greedy-based heuristic which was proposed in [19], we construct a bipartite graph that reflects the coverage of probing path and the target links. Algorithm 1 shows a greedy-based approach for the mentioned bipartite graph model that iteratively chooses the set of paths that can identify more links with smallest cost. At each iteration step, the algorithm selects a minimal solution S_i that maximizes the value of the following function:

$$\frac{\text{New Identified Links in } S_i}{\text{Cost of New Paths in } S_i}, \quad (9)$$

where the numerator is the number of uncovered identifiable links that can be covered by selecting S_i and the denominator is the cost of unselected paths in the selected set S_i .

Remark: Greedy-Min-Cost-IL is similar to the greedy heuristic proposed in [19] but with two key differences. Unlike [19] that uses uniform cost for all selected paths, we allow an arbitrary cost for each path. Furthermore, unlike [19] that requires the selected paths to cover all links, we only require the paths to identify all the identifiable links.

We use a second greedy-based approach that we don't show (due to space limitation) for the dual problem (*Min-Cost-IL*) by changing the breaking condition. The breaking condition in line 3 of algorithm 1 is changed to *while*($IL \neq I(P)$), meaning that we continue adding a new probing set S_i until all identifiable links are covered.

4.1.2 Iterative Branch-and-Bound

The ILP formulation (3, 7) allows us to apply general ILP solvers to *Max-IL-Cost* and *Min-Cost-IL*. Specifically, we use an iterative branch-and-bound algorithm [23] to achieve a configurable trade-off between complexity and optimality. For brevity, we explain the algorithm for maximization and minimization works analogously.

The algorithm first removes the integrality restrictions. The resulting linear programming (LP) relaxation of Max-IL-Cost has a polynomial time complexity and gives an upper bound (*UB*) for the maximization. If the solution satisfies all the integral constraints, then we have the optimal solution. Otherwise, we pick a

Algorithm 1: Greedy-Max-IL-Cost

Data: A set of feasible paths P , Limit on the number of paths K , Minimal combination of path sets that can identify an identifiable link l : $Z_s = \{S_l \mid l \in E\}$ where $S_l = \{r_i, \dots, r_j\}$ is the set of paths that can identify link $l \in E$

Result: A set paths $P_R \subset P$ that maximizes the number of identifiable links in $G(V, E)$, A set of identified links $IL = \{l \in E\}$

- 1 $IL = \emptyset$;
- 2 $P_R = \emptyset$;
- 3 **while** $\exists S_l \in Z_s$ **that**
- $$(K - \sum_{i=1}^{|P_R|} k_{r_i}) > (\text{Cost of New Paths in } S_l) \text{ do}$$
 - 4 Select an un-selected set
 $S_i = \arg\max \frac{\text{New Identified Links in } S_i}{\text{Cost of New Paths in } S_i};$
 - 5 for $i = 1$ to New Identified Links(S_i)
 $IL = IL \cup l \mid l \in I(S_i);$
 - 6 for $r_j \in S_i$
 $P_R = P_R \cup \{r_j\};$
 - 7
- 8 **return** IL and P_R

fractional variable, Y_r , and make two branches by creating one more constraint in the optimization: $Y_r = 0$ or $Y_r = 1$. We continue this procedure by making more branches to get closer to the optimal. The branch with the largest objective value that satisfies all the integrality constraints is called an *incumbent*. Also, at any iteration during the branch-and-bound algorithm we have a valid current upper bound, which is obtained by taking the maximum of the optimal objective values of all of the current leaf nodes. We stop branching once the gap between the incumbent's objective function (LB) and the current upper bound is smaller than a threshold (Gap), or we can stop branching after passing a given time limit. Optimality is achieved when the gap is zero. In the first case the algorithm gives a solution with an approximation ratio of $LB/(LB + Gap)$ since we have

$$\frac{LB}{OPT} \geq \frac{LB}{LB + Gap}. \quad (10)$$

In the second case, there is no guarantee on the approximation ratio but we have a guarantee on the execution time of the algorithm. Similarly, for a minimization problem (e.g., Min-Cost-IL), the incumbent (the branch with the smallest objective value and an integral solution) gives a upper bound (UB) on the optimal solution, and the LP relaxation gives a lower bound (LB). If the algorithm stops when $UB - LB \leq Gap$, then the incumbent gives a $UB/(UB - Gap)$ -approximation since



Figure 3: The iterative branch and bound algorithm that shows the gap between the incumbent and the upper bound for Max-IL-Cost.

we have

$$\frac{UB}{OPT} \leq \frac{UB}{UB - Gap}. \quad (11)$$

The advantage of this algorithm is its flexibility. We can control the stopping rule of the branch-and-bound procedure to achieve trade-off between optimality and complexity.

4.2 Algorithms for Rank Optimization

In this section, we propose two greedy-based approaches, called *Greedy-Min-Cost-Rank* for *Min-Cost-Rank* problem and *Greedy-Max-Rank-Cost* for *Max-Rank-Cost* optimization problem. We show that in terms of the rank objective, Greedy-Min-Cost-Rank provides an **optimal** solution for *Min-Cost-Rank* problem. In addition, Greedy-Max-Rank-Cost gives $1 - 1/e$ approximation for *Max-Rank-Cost* problem.

We first review the definition and properties of matroids [24] as they will prove to be useful in the remainder of the paper. Matroids play an essential role in combinatorial optimization and provide efficient and strong tool for solving computationally intractable problems.

Definition A *Matroid* is a pair $\mathcal{M} = \{L, \mathcal{I}\}$ of a finite ground set L and a collection $\mathcal{I} \subseteq 2^L$ of subsets of L such that [25, 26]:

- $\emptyset \in \mathcal{I}$
- $\forall I_x \subset I_y \subseteq L$, if $I_y \in \mathcal{I}$ then $I_x \in \mathcal{I}$
- $\forall I_x, I_y \in \mathcal{I}$, $|I_x| < |I_y| \rightarrow \exists r \in I_y \setminus I_x \text{ where } I_x \cup \{r\} \in \mathcal{I}$

We define $\mathcal{M} = \{P, \mathcal{I}\}$, where P is the set of all paths, \mathcal{I} contains the sets $P_R \subseteq P$ such that paths in P_R are linearly independent.

We are able to achieve **optimal** solution for *Min-Cost-Rank* and $1 - 1/e$ **near-optimal** approximation solution for *Max-Rank-Cost*. The first is due to the fact that the sets of linearly independent paths form a matroid, and we are selecting a basis of this matroid

with minimum cost. The approximation solution for *Max-Rank-Cost* is due to the submodularity of the rank function introduced in 3.3.2.

4.2.1 Greedy-Min-Cost-Rank

We now consider one of the interesting properties of matroids. We show that finding a maximal basis B of matroid, \mathcal{I} , of minimum weight can be solved optimally using a greedy-based heuristic. The greedy-based algorithm is similar to Kruskal's algorithm [27] that finds a minimum spanning tree in the graph. The algorithm iteratively adds a path with minimum cost to the set of selected paths until the rank of the selected paths is equal to the rank of the original routing matrix.

THEOREM 2. [24] For any routing path elements P and any probing cost function k_i , Greedy-Min-Cost-Rank (Algorithm 2) is optimal for Min-Cost-Rank, i.e., it returns a basis of P with the minimum probing cost.

Complexity Analysis: Let $F(|P|)$ be the time complexity of testing whether a ground set is independent or not (line 5-6) which is the time complexity of checking whether the rank function is increasing or not. The Greedy-Min-Cost-Rank algorithm runs in $O(|P|\log(|P|)) + |P|.F(|P|))$. Using Gaussian Elimination algorithm to compute the rank function [28], that has a time complexity of $\min(|L|, |P|) \times (|P| \times |L|)$ the complexity of the algorithm is $O(|L|^2 \times |P|^2)$, where $|P| = \frac{|M| \times (|M|-1)}{2}$.

LEMMA 4. If Greedy-Min-Cost-Rank returns a basis B for A_{*,L_I} where $\text{rank}(A_{B,L \setminus L_I}) = 0$, then B is the minimum cost set of paths that identifies all identifiable links, i.e. optimal solution to Min-Cost-IL.

PROOF. In Section 8 (APPENDIX). \square

However, if Greedy-Min-Cost-Rank returns a minimum cost basis X for A_{*,L_I} where $\text{rank}(A_{X,L \setminus L_I}) = j \neq 0$ and the selected paths' cost is K_1 , then K_1 is the lower bound for Min-Cost-Rank.

THEOREM 3. For any routing matrix A , and a set of identifiable links L_I , let Greedy-Min-Cost-Rank returns a basis $B_{A_{*,L_I}}$ for A_{*,L_I} with the minimum cost $K_{LB} = k_1 + k_2 + \dots + k_{|L_I|}$, and let Greedy-Min-Cost-Rank returns a basis B_A for the routing matrix A with the minimum cost $K_{UB} = k'_1 + k'_2 + \dots + k'_{\text{rank}(A)}$. Also let K^{opt} be the optimal cost solution of Min-Cost-IL, we have:

$$K_{LB} \leq K^{opt} \leq K_{UB} \quad (12)$$

PROOF. In Section 8 (APPENDIX). \square

Remark: Note that the difference between the lower bound K_{LB} and the upper bound K_{UB} is no larger than $k'_{|L_I|} + \dots + k'_{\text{rank}(A)}$. Since we have more constraint for

Algorithm 2: Greedy-Min-Cost-Rank approach for *Min-Cost-Rank* problem

Data: A set of paths $P = \{r_1, \dots, r_{|M|(|M|-1)/2}\}$, a set of cost functions for each path $\text{Cost} = \{k_1, \dots, k_{|M|(|M|-1)/2}\}$.

Result: A subset of paths $R^* \subseteq P$ that preserves the rank, i.e. $\text{rank}(R^*) = \text{rank}(P)$ with minimum probing cost.

```

1  $P_R^* = \emptyset;$ 
2  $TotalCost = 0;$ 
3 sort  $P$  in increasing order of cost ;
4 forall  $r_i \in P$  do
5    $IncreaseRank_{r_i} = \text{rank}(P_R^* \cup \{r_i\}) - \text{rank}(P_R^*)$ 
    ;
6   if  $IncreaseRank_{r_i} > 0$  then
7      $P_R^* = P_R^* \cup \{r_i\}$  ;
8      $TotalCost = TotalCost + k_i$  ;
9   if  $\text{rank}(P_R^*) \geq \text{rank}(P)$  or  $|I(P_R^*)| = |L_I|$  then
10    break ;
11 return  $P_R^*, TotalCost$ 

```

selecting the first $|L_I|$ paths for A_{*,L_I} than A , we always have

$$k'_1 + k'_2 + \dots + k'_{|L_I|} \leq k_1 + k_2 + \dots + k_{|L_I|} \quad (13)$$

Therefore,

$$\begin{aligned} K_{UB} - K_{LB} &\leq k'_{|L_I|} + \dots + k'_{\text{rank}(A)} \\ &\leq (\text{rank}(A) - |L_I|) * k'_{\text{rank}(A)} \end{aligned} \quad (14)$$

Tightness of the bound.

For special routing matrices, the lower or upper bound is tight and coincides with the optimal for identifiability. To prove that, we first construct a routing matrix where the lower bound is tight. For this scenario, consider a routing matrix A , where the minimum cost basis $B_{A_{*,L_I}}$ for A_{*,L_I} does not pass any of the non-identifiable links (i.e. $\text{rank}(A_{B,L \setminus L_I}) = 0$). In this scenario, the lower bound is tight and coincides with the optimal. The minimum cost basis $B_{A_{*,L_I}}$ for A_{*,L_I} returned by Greedy-Min-Cost-Rank is always optimal for Min-Cost-IL (i.e., it identifies all links in L_I with minimum cost), if $\text{rank}(A_{B_{A_{*,L_I}}, L \setminus L_I}) = 0$.

For the second scenario, we consider a network topology, where every monitor is connected to another monitor through one hop. Therefore, routing matrix is full rank and all links are identifiable. In this scenario, we need to select all paths to identify all links and thus the upper bound is tight and coincides with the optimal. The minimum cost basis B_A for A returned by Greedy-Min-Cost-Rank is always optimal for Min-Cost-IL if $\text{rank}(A) = \text{rank}(B_A) = |L_I|$.

4.2.2 Greedy-Max-Rank-Cost

Since the rank function is submodular, we can apply a modified greedy algorithm called *Greedy-Max-Rank-Cost* that gives $(1 - 1/e)$ -approximation of the *Max-Rank-Cost* problem.

Algorithm 3 shows a Greedy-Max-Rank-Cost approach that enumerates all subsets of up to 3 paths, and iteratively augments each of these subsets by adding one path at a time to maximize the increment in rank per unit cost within the probing budget. The path set with the maximum rank is then selected as the overall solution. Since the rank function is monotone and submodular (Lemma 3), we can leverage an existing result for budgeted submodular maximization.

THEOREM 4. [10] *Greedy-Max-Rank-Cost (Algorithm 3) achieves $(1 - 1/e)$ -approximation for the Max-Rank-Cost problem, i.e., the rank of its solution P is no smaller than $(1 - 1/e)$ times the maximum rank.*

Complexity Analysis: In the worst case scenario, the algorithm has to find the maximum increase of the rank function $|P|^5$ times. Therefore, the complexity of the algorithm is $O(|P|^5 F(P))$. Where $F(P)$ is the complexity of calculating rank of P . Using Gaussian Elimination algorithm to compute the rank function [28], the complexity of the algorithm is $O(|L|^2 \times |P|^6)$.

Algorithm 3 provides upper/lower bounds on the maximum identifiability that can be achieved under the given probing budget.

THEOREM 5. *Let P_R be the set of paths returned by Greedy-Max-Rank-Cost for a probing budget K , which induces a routing matrix $A_{R,*}$ and identifies I_R links. Then the maximum number of links I^{opt} that can be identified under budget K , given by the optimal solution of Max-IL-Cost, is bounded by:*

$$I_R \leq I^{opt} \leq \min\{rank(A_{R,*}) \cdot \frac{e}{e-1}, |L_I|\}, \quad (15)$$

where L_I is the set of identifiable links using all possible paths P .

PROOF. In Section 8 (APPENDIX). \square

5. EVALUATION

In this section, we consider several scenarios to compare the probing cost of our proposed algorithms compared to the case where we use all feasible probes or the optimal (OPT) brute-force solution. For each scenario, we randomize the results by running 10 different trials, where we vary the random selection of monitors from the entire set of nodes. We implement our low cost monitoring algorithms in python and used the *Gurobi* optimization toolkit, on a 120-core, 2.5 GHz, 4TB RAM cluster [29]. We assume shortest path routing (based on

Algorithm 3: Greedy-Max-Rank-Cost approach for *Max-Rank-Cost* problem

Data: A set of paths $P = \{r_1, \dots, r_{|M|(|M|-1)/2}\}$, a set of cost functions for each path $Cost = \{k_1, \dots, k_{|M|(|M|-1)/2}\}$, Limit on the probing cost K .

Result: A subset of paths $P_{Max} \subseteq P$ that maximizes $rank(P_{Max})$ subject to a limited monitoring cost K

```

1  $P_{Max2} = \emptyset;$ 
2  $TotalCost = 0;$ 
3  $P_{Max1} = argmax\{rank(P_x) : P_x \subseteq P, |P_x| \leq 3, c(P_x) \leq K\};$ 
4 forall  $P_g \subseteq P, |P_g| = 3, c(P_g) \leq K$  do
5    $P' = P \setminus P_g;$ 
6    $P = P_g;$ 
7   while  $P' \neq \emptyset$  do
8     forall  $r_i \in P'$  do
9        $IncreaseBonus_{r_i} =$ 
10       $(rank(P_{Max2} \cup \{r_i\}) - rank(P_{Max2})) / k_i$ 
11       $r_{MaxIncrease} =$ 
12       $argmax_{r_i \in P'} IncreaseBonus_{r_i}$ 
13      if  $k_{MaxIncrease} + TotalCost \leq K$  then
14         $P = P \cup \{r_{MaxIncrease}\};$ 
15         $TotalCost = k_{MaxIncrease} + TotalCost;$ 
16         $P' = P' \setminus (\{r_{MaxIncrease}\} \cup \{r \in P' : k_r > K - TotalCost\})$ 
17
18 if  $rank(P) > rank(P_{max2})$  then
19    $P_{Max2} = P$ 

```

18 return $argmax_{P \in \{P_{Max1}, P_{Max2}\}} rank(P)$

Table 3: Network characteristics used in our evaluation.

Network Name	# of nodes	# of edges	Average Node degree
Abilene	11	14	2.5
BellCanada	48	64	2.62
CAIDA	825	1018	2.46

hop count), with ties broken arbitrarily.

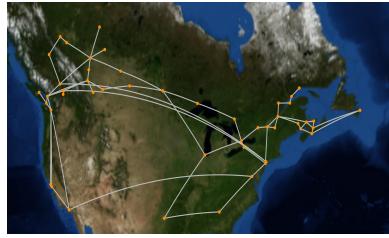
We use different network topologies including a small (Abilene) and a medium topology (BellCanada) taken from the Internet Topology Zoo [20, 21]. We also consider AS28717 topology (CAIDA) taken from the CAIDA (Center for Applied Internet Data Analysis) dataset [30]. The network topologies used in our evaluation are shown in Figure 4. Table 3 shows the characteristics of these topologies.

5.1 Identifiability Maximization

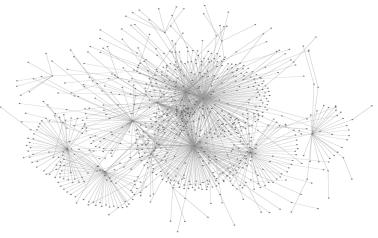
In the first set of simulations, we consider the impact of probing cost limit on the number of identifiable links. We first compare the upper and lower bound of Theorem 5 with maximum number of identifiable links



(a) Abilene.

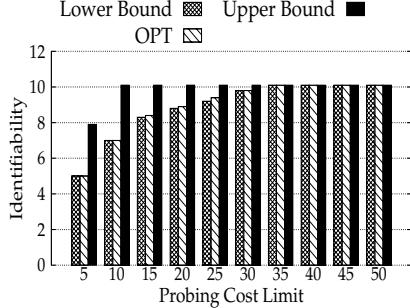


(b) BellCanada.

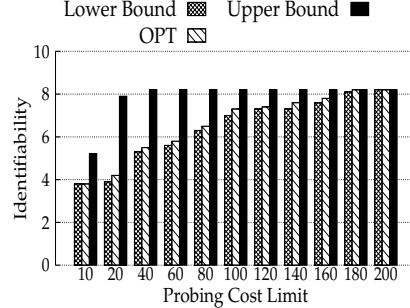


(c) CAIDA.

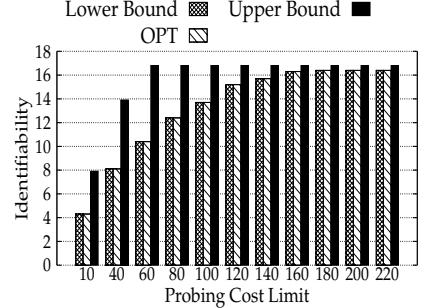
Figure 4: Network topology of graphs used in the evaluation a) Abilene, b) BellCanada and c) CAIDA topology.



(a) Abilene.

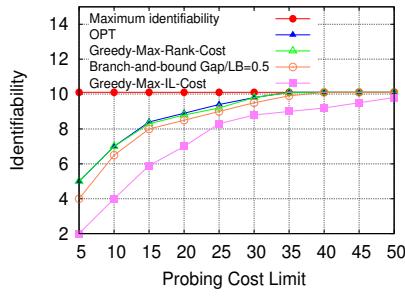


(b) BellCanada.

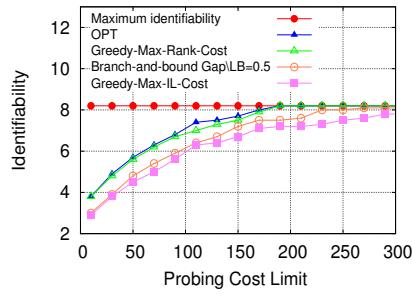


(c) CAIDA.

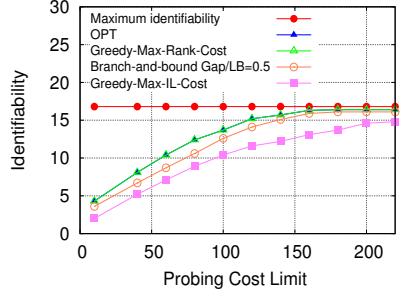
Figure 5: Upper and lower bound on the number of identifiable links as a function of limit on the probing cost in a) Abilene (9 monitors), b) BellCanada (10 monitors) and c) CAIDA topology (9 monitors).



(a) Abilene.

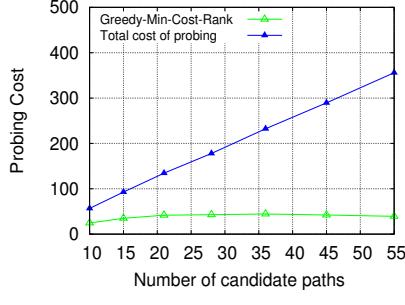


(b) BellCanada.

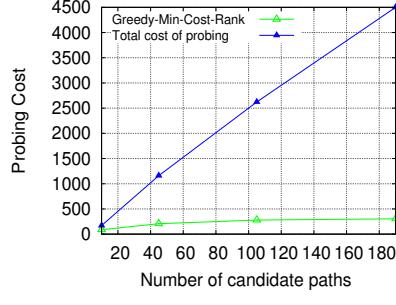


(c) CAIDA.

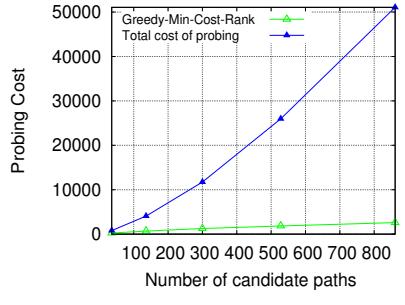
Figure 6: Number of identifiable links as a function of limit on the probing cost in a) Abilene (9 monitors), b) BellCanada (10 monitors) and c) CAIDA topology (9 monitors).



(a) Abilene.



(b) BellCanada.



(c) CAIDA.

Figure 7: Probing cost vs number of feasible probing paths in a) Abilene (5-11 monitors), b) BellCanada (10-29 monitors) and c) CAIDA (9-42 monitors) topology.

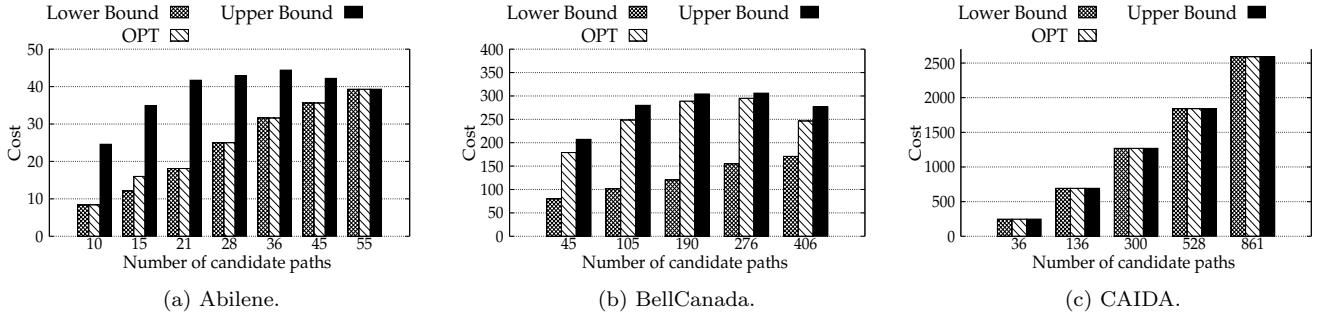


Figure 8: Upper and lower bound on the probing cost as a function of number of candidate paths in a) Abilene (5-11 monitors), b) BellCanada (10-29 monitors) and c) CAIDA (9-42 monitors) topology.

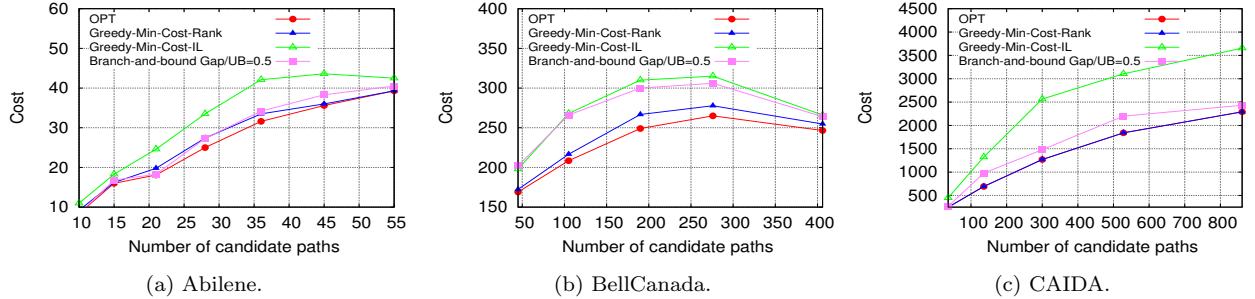


Figure 9: Number of identifiable links as a function of limit on the probing cost in a) Abilene (5-11 monitors), b) BellCanada (10-29 monitors) and c) CAIDA (9-42 monitors) topology.

using all candidate paths. We randomly select 9, 10 and 9 monitors from the entire set of nodes in Abilene, BellCanada and CAIDA topologies respectively. Figures 5a, 5b and 5c show the lower and upper bound on the number of identifiable links and the optimal number of identifiable links for each topology. We note that the optimal number of identifiable links is always upper bounded by the minimum of (i) maximum identifiability and (ii) the upper bound in Theorem 5. As shown, the difference between the optimal number of identifiable links in the upper and lower bound is very small which shows that Greedy-Max-Rank-Cost gives a solution close to optimal. Further, we note that the lower bound is closer to the optimal and gives a tighter bound in terms of number of identifiable links.

We next compare the number of identifiable links in Greedy-Max-IL-Cost heuristic (Algorithm 1), Greedy-Max-Rank, and the optimal case (OPT). We use gurobi optimization toolkit to solve the ILP problem formulation (equation 2). We also use our iterative branch-and-bound algorithm and stop the search when $Gap \leq 0.5 \cdot LB$. We recall that, the larger the gap is, the lower is the number of iterations of the optimization algorithm is and therefore we have an approximation of the solution which is farther from optimal. Figures 6a, 6b and 6c show scenarios where we increase the limit on the probing cost of monitors for the Abilene, BellCanada and CAIDA topology. As we increase the probing cost

limit, more links are uniquely identified and all algorithm eventually converge to maximum identifiability, while Greedy-Max-Rank is closest to the optimal.

5.2 Cost Minimization

In the next set of simulations, we evaluate the performance of *Greedy-Min-Cost-Rank* algorithm that preserves rank and the greedy-based heuristics that preserve identifiability. We consider Abilene, BellCanada and CAIDA topology and run *Greedy-Min-Cost-Rank* and *Greedy-Min-Cost-IL* algorithms that preserve rank and identifiability respectively. We also run our branch-and-bound formulation and stop branching once $Gap \leq 0.5 \cdot UB$. In the first set of experiments, we increase the number of candidate paths by increasing the number of randomly selected monitors and evaluate the cost saving of our *Greedy-Min-Cost-Rank* algorithm with respect to the case where we use all candidate paths. By choosing M random monitors, we have $\frac{M \times (M-1)}{2}$ candidate paths. We increase the number monitors from 5 to 11 in Abilene, 10 to 29 in BellCanada and 9 to 42 in CAIDA topology. Figures 7a, 7b and 7c show the simulation results for this scenario. As shown, probing all candidate paths generates a large amount of traffic and incurs a high cost, while our *Greedy-Min-Cost-Rank* algorithm significantly reduces the cost. We also compare the accuracy of *Greedy-Min-Cost-Rank* by running the algorithm on (i) the entire routing matrix A , and (ii)a

subset of the routing matrix with columns corresponding to the set of identifiable links A_{*,L_I} ; the former gives the upper bound and the latter gives a lower bound on the probing cost according to Theorem 3. Figure 8 shows the upper and lower bound on the probing cost as we increase the number of candidate paths in each network topology. The simulation results for CAIDA topology, shows that the number of identifiable links is equal to the rank of the routing matrix A and thus the upper bound and lower bound are equal. Therefore, the solution to Greedy-Min-Cost-Rank for this topology is optimal. In Abilene and BellCanada topology, the lower bound and upper bound are closer to the optimal respectively.

We next evaluate the probing cost of each algorithm compared to optimal. Figures 9a, 9b and 9c show the probing cost of each network topology as we increase the number of candidate paths. As shown, our Greedy-Min-Cost-Rank algorithm is closer to the optimal in all topologies and coincides with the optimal in CAIDA.

6. CONCLUSION

This paper studies the optimal selection of monitoring paths to balance identifiability and cost. We consider the constrained optimization problem of 1) maximizing identifiability under limited probing budget, 2) maximizing the rank function under a limited probing budget, 3) minimizing the probing cost subject to preserving identifiability, and 4) minimizing the probing cost subject to preserving the rank. While (1) and (3) are hard to solve, (2) and (4) possess desirable properties that allow efficient computation while providing good approximation to (1) and (3). We proposed an optimal greedy-based approach for (4) and proposed a $(1 - 1/e)$ -approximation algorithm for (2). Our experimental analysis reveals that, compared to several greedy approaches, our rank-based optimization performs better in terms of identifiability and probing cost. Furthermore, our solution can reduce the total probing cost by an order of magnitude while achieving the same monitoring performance.

Acknowledgement

This research is supported by the Defense Threat Reduction Agency under Grant HDTRA1-10-1-0085, and is built upon research sponsored by the U.S. Army Research Laboratory and the U.K. Ministry of Defense under Agreement Number W911NF-06-3-0001.

7. REFERENCES

- [1] A. Adams et al. The use of end-to-end multicast measurements for characterizing internal network behavior. *IEEE Communications magazine*, 2000.
- [2] Y. Vardi. Network tomography: Estimating source-destination traffic intensities from link data. *Journal of the American statistical association*, 1996.
- [3] D. Z. Tootaghaj et al. Network recovery from massive failures under uncertain knowledge of damages. In *the Proceedings of the IFIP Networking Conference (IFIP NETWORKING)*, 2017.
- [4] D. Z. Tootaghaj et al. Controlling cascading failures in interdependent networks under incomplete knowledge. In *The 36th IEEE International Symposium on Reliable Distributed Systems (SRDS)*, 2017.
- [5] W. Stallings. Snmp and snmpv2: the infrastructure for network management. *IEEE Communications Magazine*, 1998.
- [6] D. Andersen et al. *Resilient overlay networks*. ACM, 2001.
- [7] N. Feamster et al. Measuring the effects of internet path faults on reactive routing. In *ACM SIGMETRICS Performance Evaluation Review*. ACM, 2003.
- [8] V. Paxson. End-to-end routing behavior in the internet. *IEEE/ACM transactions on Networking*, 1997.
- [9] M. Luckie et al. Challenges in inferring internet interdomain congestion. In *IMC*. ACM, 2014.
- [10] S. Khuller et al. The budgeted maximum coverage problem. *Information Processing Letters*, 1999.
- [11] D. S. Hochbaum. Approximation algorithms for the set covering and vertex cover problems. *SIAM Journal on computing*, 1982.
- [12] Y. Bejerano et al. Robust monitoring of link delays and faults in ip networks. *IEEE/ACM Transactions on Networking (TON)*, 2006.
- [13] R. Kumar et al. Practical beacon placement for link monitoring using network tomography. *IEEE Journal on Selected Areas in Communications*, 2006.
- [14] A. Gopalan et al. On identifying additive link metrics using linearly independent cycles and paths. *IEEE/ACM Transactions on Networking (TON)*, 2012.
- [15] L. Ma et al. Inferring link metrics from end-to-end path measurements: Identifiability and monitor placement. *IEEE/ACM Transactions on Networking (TON)*, 2014.
- [16] L. Ma et al. Efficient identification of additive link metrics via network tomography. In *IEEE ICDCS*, 2013.
- [17] Y. Chen et al. An algebraic approach to practical and scalable overlay network monitoring. In *ACM SIGCOMM Computer Communication Review*. ACM, 2004.
- [18] F. Li et al. End-to-end service quality measurement using source-routed probes. In *INFOCOM*, 2006.
- [19] Q. Zheng et al. Minimizing probing cost and achieving identifiability in probe-based network link monitoring. *IEEE Transactions on Computers*, 2013.
- [20] S. Knight et al. The internet topology zoo. *IEEE Journal on Selected Areas in Communications*, 2011.
- [21] The internet topology zoo. <http://www.topology-zoo.org/>, accessed in May, 2015.
- [22] S. Fujishige. *Submodular functions and optimization*. Elsevier, 2005.
- [23] G. L. Nemhauser et al. Integer programming and combinatorial optimization. *Constraint Classification for Mixed Integer Programming Formulations*. COAL Bulletin, 1988.
- [24] J. G. Oxley. *Matroid theory*, volume 3. Oxford University Press, USA, 2006.
- [25] K. Murota. *Matrices and matroids for systems analysis*, volume 20. Springer Science & Business Media, 2009.
- [26] T. He et al. Service placement for detecting and localizing failures using end-to-end observations. In *IEEE ICDCS*, 2016.
- [27] J. B. Kruskal. On the shortest spanning subtree of a graph and the traveling salesman problem. *Proceedings of the American Mathematical society*, 1956.
- [28] G. Strang. Introduction to linear algebra. 2011.
- [29] Gurobi optimization, inc. gurobi optimizer reference manual. <http://www.gurobi.com/>, accessed in, 2012.
- [30] The cooperative association for internet data analysis (caida), macroscopic internet topology data kit (itdk). <http://www.caida.org>, 2013.

8. APPENDIX

PROOF OF THEOREM 1. We first show the necessary condition, i.e. if a set of routing paths $P_R \subseteq P$ identifies all identifiable links in L_1 , then $\text{rank}(A_{R,*}) = \text{rank}(A_{R,L_1}) + \text{rank}(A_{R,L \setminus L_1})$ and $\text{rank}(A_{R,L_1}) = \text{rank}(A_{*,L_1}) = |L_1|$. Suppose that the routing matrix $A_{R,*}$ is of size $n \times |L|$.

Without loss of generality (WLOG), suppose that the number of identifiable links in L_1 is $|L_1| = k$ and the first k columns of A_R correspond to these k identifiable links (one can exchange the columns in A_R to have this property). This means that the reduced row echelon form of $A_{R,*}$ should be as follows:

$$\text{rref}(A_{R,*}) = \left[\begin{array}{c|c} I_{k \times k} & 0_{k \times (|L|-k)} \\ \hline 0_{(n-k) \times k} & M_{(n-k) \times (|L|-k)} \end{array} \right] \quad (16)$$

Where, $I_{k \times k}$ is the identity matrix and $0_{k \times (|L|-k)}$ and $0_{(n-k) \times |L|}$ are matrices containing all zero entries and $M_{(n-k) \times (|L|-k)}$ is a matrix of general values. Therefore, the rank of $A_{R,*}$ is as follows:

$$\text{rank}(A_{R,*}) = k + \text{rank}(M) \quad (17)$$

It is clear that:

$$\begin{aligned} \text{rank}(A_{R,L_1}) &= k, \\ \text{rank}(A_{R,L \setminus L_1}) &= \text{rank}(M) \end{aligned} \quad (18)$$

Therefore,

$$\begin{aligned} \text{rank}(A_{R,*}) &= k + \text{rank}(M) = \\ &= \text{rank}(A_{R,L_1}) + \text{rank}(A_{R,L \setminus L_1}) \end{aligned} \quad (19)$$

Next, we prove the sufficient condition, i.e. if for a selected subset of paths $P_R \subseteq P$ (1) is satisfied, then, P_R can solve all identifiable links.

Since $\text{rank}(A_R) \leq \text{rank}(A_{R,L_1}) + \text{rank}(A_{R,L \setminus L_1})$, and $\text{rank}(A_{R,L_1}) \leq |L_1|$, (1) implies that $\text{rank}(A_{R,L_1}) = |L_1|$, i.e., rows of A_{R,L_1} contain a basis of the row space of A_{*,L_1} . Therefore the reduced row echelon form of $A_{R,*}$ should contain the identity matrix $I_{k \times k}$ as follows:

$$\text{rref}(A_{R,*}) = \left[\begin{array}{c|c} I_{k \times k} & B_{k \times (|L|-k)} \\ \hline C_{(n-k) \times k} & M_{(n-k) \times (|L|-k)} \end{array} \right] \quad (20)$$

We show that the submatrices $B_{k \times (|L|-k)}$ and $C_{(n-k) \times k}$ must be zero matrices. If $C_{(n-k) \times k}$ contains a non-zero entry, we can make them zero by using a sequence of elementary row operations. Note that (1) implies that

$$\begin{aligned} \text{rank}(\text{rref}(A_{R,*})) &= \text{rank}\left(\left[\begin{array}{c} I_{k \times k} \\ 0_{(n-k) \times k} \end{array} \right]\right) + \\ &\quad \text{rank}\left(\left[\begin{array}{c} B_{k \times (|L|-k)} \\ M_{(n-k) \times (|L|-k)} \end{array} \right]\right) \end{aligned} \quad (21)$$

To prove that $B_{k \times (|L|-k)} = 0$, we re-write the reduced

row echelon form of $A_{R,*}$ as follows:

$$\begin{aligned} \text{rref}(A_{R,*}) &= \left[\begin{array}{c|c} I_{k \times k} & B_{k \times (|L|-k)} \\ \hline 0_{(n-k) \times k} & M_{(n-k) \times (|L|-k)} \end{array} \right] \\ &= \left[\begin{array}{c|ccc} I_{k \times k} & b_1 & \dots & b_{|L|-k} \\ \hline 0_{(n-k) \times k} & m_1 & \dots & m_{|L|-k} \end{array} \right], \end{aligned} \quad (22)$$

where, b_i and m_i are the i -th column of $B_{k \times (|L|-k)}$ and $M_{(n-k) \times (|L|-k)}$ respectively. Let $[e_1, e_2, \dots, e_k]$ be the columns of $\left[\begin{array}{c} I_{k \times k} \\ 0_{(n-k) \times k} \end{array} \right]$. Also let $[q_1, q_2, \dots, q_{|L|-k}]$ be the columns of $\left[\begin{array}{c} B \\ M \end{array} \right]$, where $q_i = \left[\begin{array}{c} b_i \\ m_i \end{array} \right]$.

We define the indicator functions δ_i and δ'_i as follows:

$$\delta_i = \begin{cases} 1, & \text{if } q_i \text{ is independent of } \{e_1, \dots, e_k\} \cup \{q_1, \dots, q_{i-1}\} \\ 0, & \text{Otherwise.} \end{cases}$$

$$\delta'_i = \begin{cases} 1, & \text{if } q_i \text{ is independent of } \{q_1, \dots, q_{i-1}\} \\ 0, & \text{Otherwise.} \end{cases}$$

LEMMA 5. *We claim that*

$$\delta_i = \delta'_i \quad \text{for } i = 1, \dots, |L|-k, \quad (23)$$

i.e., q_i is linearly independent of $\{e_1, \dots, e_k\} \cup \{q_1, \dots, q_{i-1}\}$, if q_i is linearly independent of $\{q_1, \dots, q_{i-1}\}$.

To see this, we note that the left hand side (LHS) and right hand side of Equation (21) are as follows:

$$\text{LHS of (21): } \text{rank}(\text{rref}(A_R)) = k + \sum_{i=1}^{|L|-k} \delta_i \quad (24)$$

$$\text{RHS of (21): } \text{rank}(\text{rref}(A_R)) = k + \sum_{i=1}^{|L|-k} \delta'_i \quad (25)$$

It is clear that $\delta_i \leq \delta'_i$, $\forall i = 1, \dots, |L|-k$, because if q_i is linearly independent of $\{q_1, \dots, q_{i-1}\} \cup \{e_1, \dots, e_k\}$ it has to be independent of $\{q_1, \dots, q_{i-1}\}$ which is a subset of the former. Thus, if $\exists i \in \{1, \dots, |L|-k\}$ such that $\delta_i < \delta'_i$, (24) will be smaller than (25), violating Equation (21). Thus, $\delta_i = \delta'_i, \forall i = 1, \dots, |L|-k$. Using the above claim, we prove $b_i = 0$, $i = 1, \dots, |L|-k$ by induction. For $i = 1$, if row $k+1$ in $\text{rref}(A_{R,*})$ contains a pivot in column $k+1$ (i.e. q_1 contains a pivot), then by definition of the reduced row echelon form, other entries in column $k+1$ should be zero and thus $b_1 = 0$. If row $k+1$ in $\text{rref}(A_{R,*})$ does not contain a pivot in column $k+1$ (i.e., not contain a pivot or contain a pivot in column $j > k+1$), then the non-zero entries (if any) in row $k+1$ and every row below row $k+1$ must be to the right of column $k+1$, i.e. $m_1 = 0$. Therefore, $q_1 = \left[\begin{array}{c} b_1 \\ m_1 \end{array} \right]$ is linearly dependent with $\{e_1, \dots, e_k\}$. By

(23), $q_1 = 0$ and thus $b_1 = 0$.

For $i > 1$, assume $b_j = 0$ for $j = 1, \dots, i-1$. If $q_i = \begin{bmatrix} b_i \\ m_i \end{bmatrix}$ contains a pivot, then the pivot must be in a row below row k (as (22) already indicates that the pivots in rows $1, \dots, k$ appear before column q_i). Thus by definition of reduced row echelon form $b_i = 0$. If q_i does not contain a pivot, then q_i can be written as linear combination of $\{e_1, \dots, e_k\}$ and $\{q_{i_l}\}_{l=1}^i$ where i_l is the index for those columns in $\{q_1, \dots, q_i\}$ which contain a pivot. Thus, q_i is linearly dependent of $\{e_1, \dots, e_k\} \cup \{q_1, \dots, q_{i-1}\}$. By Lemma 5, q_i is linearly dependent of $\{q_1, \dots, q_{i-1}\}$. Since $b_j = 0$ for $j = 1, \dots, i-1$, b_i must be zero.

Therefore, using the reduced row echelon form of $A_{R,*}$ in (20), each link in L_1 , corresponding to one of the first k columns in $rref(A_R)$, can be uniquely determined from the set of selected paths P_R . We therefore, conclude that the necessary and sufficient condition for $P_R \subseteq P$ to identify a set of links L_1 is $rank(A_{R,*}) = |L_1| + rank(A_{R,L \setminus L_1})$. \square

PROOF OF LEMMA 2. Suppose there exists an optimal solution of the LP-relaxation of Max-IL-Cost over Z_s and X_l , where $\exists l \in L$ with $0 < X_l < 1$. Therefore, $\exists s : l \in I(P_s)$ such that $Z_s > 0$. From (3-d), it implies that if $Z_s > 0$, we must have $Y_r = 1 \quad \forall r \in P_s$. Therefore, we can make $Z_s = 1$, and $X_l = 1$ to increase the value of the objective function without violating any constraint. This contradicts with the assumption that this solution is optimal. Similar argument shows a contradiction if $\exists s \in S \text{ s.t. } 0 < Z_s < 1$. Therefore, the optimal solution of the LP relaxation over Z_s and X_l

always gives an integer solution. \square

PROOF OF LEMMA 4. A path set identifies all the identifiable links if and only if it satisfies the conditions of Theorem 1, i.e. $rank(A_{R,*}) = |L_I| + rank(A_{R,L \setminus L_I})$. Note that R is a solution to Min-Cost-Rank since we know $rank(A_{R,*}) = |L_I|$. Thus, the optimal solution to Min-Cost-IL is identical to the optimal solution to Min-Cost-Rank, given by Greedy-Min-Cost-Rank by theorem 2. \square

PROOF OF THEOREM 3. The lower bound is obvious, since we showed that *Greedy-Min-Cost-Rank* returns the **optimal** minimum basis for A_{*,L_I} , there is no lower cost set of paths that is both a basis for A_{*,L_I} and satisfies the conditions of theorem 1.

For the upper bound, note that any basis B_A for the routing matrix A identifies all links in L_I and thus has lower cost than K^{opt} . \square

PROOF OF THEOREM 5. The lower bound $I_R \leq I^{opt}$ trivially holds due to the optimality of I^{opt} . For the upper bound, we denote by R^* the set of path indices in the optimal solution of Max-IL-Cost. Then by Theorem 1, $I^{opt} \leq rank(A_{R^*})$. Meanwhile, by Theorem 4, we have that

$$rank(A_{R^*}) \leq rank^{opt} \leq rank(A_R) \cdot \frac{e}{e-1}, \quad (26)$$

where $rank^{opt}$ is the rank of the optimal solution of Max-Rank-Cost. This gives the upper bound on I^{opt} . Also, note that I^{opt} is always smaller than the maximum identifiability ($|L_I|$) using all possible paths in P . \square