Network Capability in Localizing Node Failures via End-to-end Path Measurements

Liang Ma[†], Ting He[†], Ananthram Swami[§], Don Towsley[*], and Kin K. Leung[‡]
[†]IBM T. J. Watson Research Center, Yorktown, NY, USA. Email: {maliang, the}@us.ibm.com
[§]Army Research Laboratory, Adelphi, MD, USA. Email: ananthram.swami.civ@mail.mil
[*]University of Massachusetts, Amherst, MA, USA. Email: towsley@cs.umass.edu
[‡]Imperial College, London, UK. Email: kin.leung@imperial.ac.uk

***Abstract*—We investigate the capability of localizing node failures in communication networks from binary states (normal/failed) of end-to-end paths. Given a set of nodes of interest, uniquely localizing failures within this set requires that different observable path states associate with different node failure events. However, this condition is difficult to test on large networks due to the need to enumerate all possible node failures. Our first contribution is a set of sufficient/necessary conditions for identifying a bounded number of failures within an arbitrary node set that can be tested in polynomial time. In addition to network topology and locations of monitors, our conditions also incorporate constraints imposed by the probing mechanism used. We consider three probing mechanisms that differ according to whether measurement paths are (i) arbitrarily controllable, (ii) controllable but cycle-free, or (iii) uncontrollable (determined by the default routing protocol). Our second contribution is to quantify the capability of failure localization through (1) the maximum number of failures (anywhere in the network) such that failures within a given node set can be uniquely localized, and (2) the largest node set within which failures can be uniquely localized under a given bound on the total number of failures. Both measures in (1–2) can be converted into functions of a per-node property, which can be computed efficiently based on the above sufficient/necessary conditions. We demonstrate how measures (1–2) proposed for quantifying failure localization capability can be used to evaluate the impact of various parameters, including topology, number of monitors, and probing mechanisms.**

***Index Terms*—Network Tomography, Failure Localization, Identifiability Condition, Maximum Identifiability Index**

I. Introduction

Effective monitoring of network performance is essential for network operators in building reliable communication networks that are robust to service disruptions. In order to achieve this goal, the monitoring infrastructure must be able to detect network misbehaviors (e.g., unusually high loss/latency, unreachability) and localize the sources of the anomaly (e.g., malfunction of certain routers) in an accurate and timely manner. Knowledge of where problematic network elements reside in the network is particularly useful for fast service recovery, e.g., the network operator can migrate affected services and/or reroute traffic. However, localizing network elements that cause a service disruption can be challenging. The straightforward approach of directly monitoring the health of individual elements is not always feasible due to traffic overhead, access control, or lack of protocol support at internal nodes. Moreover, built-in monitoring agents running on network elements cannot detect problems caused by misconfigured/unanticipated interactions between network layers, where end-to-end communication is disrupted but individual network elements along the path remain functional (a.k.a. *silent failures*) [1]. These limitations call for a different approach that can diagnose the health of network elements from the health of end-to-end communications perceived between measurement points.

One such approach, generally known as *network tomography* [2], focuses on inferring internal network characteristics based on end-to-end performance measurements from a subset of nodes with monitoring capabilities, referred to as *monitors*. Unlike direct measurement, network tomography only relies on end-to-end performance (e.g., path connectivity) experienced by data packets, thus addressing issues such as overhead, lack of protocol support, and silent failures. In cases where the network characteristic of interest is binary (e.g., *normal* or *failed*), this approach is known as *Boolean network tomography* [3].

In this paper, we study an application of Boolean network tomography to localize node failures from measurements of path states[1]. Under the assumption that a measurement path is normal if and only if all nodes on this path behave normally, we formulate the problem as a system of Boolean equations, where the unknown variables are the binary node states, and the known constants are the observed states of measurement paths. The goal of Boolean network tomography is essentially to solve this system of Boolean equations.

Because the observations are coarse-grained (path normal/failed), it is usually impossible to uniquely identify node states from path measurements. For example, if two nodes always appear together in measurement paths, then upon observing failures of all these paths, we can at most deduce that one of these nodes (or both) has failed but

Research was sponsored by the U.S. Army Research Laboratory and the U.K. Ministry of Defence and was accomplished under Agreement Number W911NF-06-3-0001. The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the U.S. Army Research Laboratory, the U.S. Government, the U.K. Ministry of Defence or the U.K. Government. The U.S. and U.K. Governments are authorized to reproduce and distribute reprints for Government purposes notwithstanding any copyright notation hereon.

[1]This model can also capture link failures by transforming the topology into a logical topology with each link represented by a virtual node connected to the nodes incident to the link.

cannot determine which one. Because there are often multiple explanations for given path failures, existing work mostly focuses on finding the minimum set of failed nodes that most probably involves failed nodes. Such an approach, however, does not guarantee that nodes in this minimum set have failed or that nodes outside the set have not. Generally, to distinguish between two possible failure sets, there must exist a measurement path that traverses one and only one of these two sets. There is, however, a lack of understanding of what this requires in terms of observable network properties such as topology, monitor placement, and measurement routing. On the other hand, even if there exists ambiguity in failure localization across the entire network, it is still possible to uniquely localize node failures in a specific sub-network (e.g., sub-network with a large fraction of monitors). To determine such unique failure localization in sub-networks, we need to understand how it is related to network properties.

In this paper, we consider three closely related problems: Let S denote a set of nodes of interest (i.e., there can be ambiguity in determining the states of nodes outside S; however, the states of nodes in S must be uniquely determinable). (1) If the number of simultaneous node failures is bounded by k, then under what conditions can one uniquely localize failed nodes in S from path measurements available in the entire network? (2) What is the maximum number of simultaneous node failures (i.e., the largest value of k) such that any failures within S can be uniquely localized? (3) What is the largest node set within which failures can be uniquely localized, if the total number of failures is bounded by k? Answers to questions (2) and (3) together quantify a network's capability to localize failures from end-to-end measurements: question (2) characterizes the *scale* of failures and question (3) the *scope* of localization. Clearly, answers to the above questions depend on which paths are measurable, which in turn depends on network topology, placement of monitors, and the routing mechanism of probes. We will study all these problems in the context of the following classes of probing mechanisms: (i) *Controllable Arbitrary-path Probing (CAP)*, where any measurement path can be set up by monitors, (ii) *Controllable Simple-path Probing (CSP)*, where any measurement path can be set up, provided it is cycle-free, and (iii) *Uncontrollable Probing (UP)*, where measurement paths are determined by the default routing protocol. These probing mechanisms assume different levels of control over routing of probing packets and are feasible in different network scenarios (see Section II-C); answers to the above three problems under these probing mechanisms thus provide insights on how the level of control bestowed on the monitoring system affects its capability in failure localization.

A. Related Work

Existing work can be broadly classified into single failure localization and multiple failure localization. Single failure localization assumes that multiple simultaneous failures happen with negligible probability. Under this assumption, [4], [5] propose efficient algorithms for monitor placement such that any single failure can be detected and localized. To improve the resolution in characterizing failures, range tomography in [6] not only localizes the failure, but also estimates its severity (e.g., congestion level). These works, however, ignore the fact that multiple failures occur more frequently than one may imagine [7]. In this paper, we consider the general case of localizing multiple failures.

Multiple failure localization faces inherent uncertainty. Most existing works address this uncertainty by attempting to find the minimum set of network elements whose failures explain the observed path states. Under the assumption that failures are low-probability events, this approach generates the most probable failure set among all possibilities. Using this approach, [8], [9] propose solutions for networks with tree topologies, which are later extended to general topologies in [1]. Similarly, [10] proposes to localize link failures by minimizing false positives; however, it cannot guarantee unique failure localization. In a Bayesian formulation, [11] proposes a two-stage solution which first estimates the failure (loss rate above threshold) probabilities of different links and then infers the most likely failure set for subsequent measurements. By augmenting path measurements with (partially) available control plane information (e.g., routing messages), [12], [13] propose a greedy heuristic for troubleshooting network unreachability in multi-AS (Autonomous System) networks that has better accuracy than benchmarks using only path measurements.

Little is known when we insist on *uniquely* localizing network failures. Given a set of monitors known to uniquely localize failures on paths between themselves, [14] develops an algorithm to remove redundant monitors such that all failures remain identifiable. If the number of failed links is upper bounded by k and the monitors can probe arbitrary cycles or paths containing cycles, [15] proves that the network must be $(k+2)$-edge-connected to identify any failures up to k links using one monitor, which is then used to derive requirements on monitor placement for general topologies. Solving node failure localization using the results of [15], however, requires a topology transformation that maps each node to a link while maintaining adjacency between nodes and feasibility of measurement paths. To our knowledge, no such transformation exists whose output satisfies the assumptions of [15] (undirected graph, measurement paths not containing repeated links). Later, [16] proves that under a CAP-like probing mechanism, the condition can be relaxed to the network being k-edge-connected. Both [15], [16] focus on placing monitors and constructing measurement paths to localize a given number of failures; in contrast, we focus on characterizing the capability of failure localization under a given monitor placement and constraints on measurement paths. In previous work [17], we propose efficient testing conditions and algorithms to quantify the capability of localizing node failures in the entire network; however, we did not consider the case that even if some node states cannot be uniquely determined, we may still be able to unambiguously determine the states of some other nodes. In this paper, we thus investigate the relationships between the capability of localizing node failures and explicit network properties such as topology, placement of monitors, probing mechanism, and nodes of interest, with focus on developing efficient algorithms to characterize the capability under given settings.

A related but fundamentally different line of work is graph-

constrained group testing [18], which studies the minimum number of measurement paths needed to uniquely localize a given number of (node/link) failures, using a CAP-like probing mechanism. In contrast, we seek to characterize the type of failures (number and location) that can be uniquely localized using a variety of probing mechanisms.

B. Summary of Contributions

We study the fundamental capability of a network with arbitrarily placed monitors to uniquely localize node failures from binary end-to-end measurements between monitors. Our contributions are five-fold:

1) We propose two novel measures to quantify the capability of failure localization, (i) *maximum identifiability index* of a given node set, which characterizes the maximum number of simultaneous failures such that failures within this set can be uniquely localized, and (ii) *maximum identifiable set* for a given upper bound on the number of simultaneous failures, which represents the largest node set within which failures can be uniquely localized if the failure event satisfies the bound. We show that both measures can be expressed as functions of per-node maximum identifiability index (i.e., maximum number of failures such that the failure of a given node can be uniquely determined).

2) We establish necessary/sufficient conditions for uniquely localizing failures in a given set under a bound on the total number of failures, which are applicable to all probing mechanisms. We then convert these conditions into more concrete conditions in terms of network topology and placement of monitors, under the three different probing mechanisms (CAP, CSP, and UP), which can be tested in polynomial time.

3) We show that a special relationship between the above necessary/sufficient conditions leads to tight upper/lower bounds on the maximum identifiability index of a given set that narrows its value to at most two consecutive integers. These conditions also enable a strategy for constructing inner/outer bounds (i.e., subset/superset) of the maximum identifiable set. These bounds are polynomial-time computable under CAP and CSP. While they are NP-hard to compute under UP, we present a greedy heuristic to compute a pair of relaxed bounds that frequently coincide with the original bounds in practice.

4) We evaluate the proposed measures under different probing mechanisms on random and real topologies. Our evaluation shows that controllable probing, especially CAP, significantly improves the capability of node failure localization over uncontrollable probing. Our result also reveals novel insights into the distribution of per-node maximum identifiability index and its relationship with graph-theoretic node properties.

Note: Our results are also applicable to transient failures as long as node failures persist during probing (i.e., leading to failures of all traversing paths). We have limited our observations to binary states (normal/failed) of measurement paths. It is possible in some networks to obtain extra information from probes, e.g., rerouted paths after a default path fails, in which case our solution provides lower bounds on the capability of localizing failures. Furthermore, we do not make any assumption on the distribution or correlation of node failures across the network. In some application scenarios (e.g., datacenter networks), node failures may be correlated (e.g., all routers sharing the same power/chiller). We leave the characterization of failure localization in the presence of such additional information to future work.

TABLE I
GRAPH-RELATED NOTATIONS

Symbol	Meaning
V, L	set of nodes/links ($\xi := \lvert L\rvert$)
M, N	set of monitors/non-monitors ($M \cup N = V$, $\mu := \lvert M\rvert$, $\sigma := \lvert N\rvert$)
k	maximum number of simultaneous non-monitor failures
$V(\mathcal{G})$	set of nodes in $\mathcal{G}$
$\mathcal{N}(M)$	set of non-monitors that are neighbors of at least one monitor in M ($\theta := \lvert\mathcal{N}(M)\rvert$)
$\mathcal{L}(V, W)$	$\mathcal{L}(V, W) = \{$link vw : $\forall v \in V$, $w \in W$, $v \neq w\}$
$\mathcal{G} - L'$	delete links: $\mathcal{G} - L' = (V, L \setminus L')$, where "$\setminus$" is setminus
$\mathcal{G} + L'$	add links: $\mathcal{G} + L' = (V, L \cup L')$, where the end-points of links in L' must be in V
$\mathcal{G} - V'$	delete nodes: $\mathcal{G} - V' = (V \setminus V', L \setminus L(V'))$, where $L(V')$ is the set of links incident to nodes in V'
$\mathcal{G} + V'$	add nodes: $\mathcal{G} + V' = (V \cup V', L)$
$\mathcal{G}^*$	auxiliary graph of $\mathcal{G}$ (see Fig. 2)
$\mathcal{G}_m$	auxiliary graph of $\mathcal{G}$ w.r.t. monitor m (see Fig. 2)
$\mathcal{G}'$	extended graph of $\mathcal{G}$ (see Fig. 3)
$\Omega(S)$, $\Omega(v)$	maximum identifiability index of S or v (S: a set of nodes, v: a node)
$S^*(k)$	maximum k-identifiable set
$S^{\text{inner}}(k)$	subset of $S^*(k)$
$S^{\text{outer}}(k)$	superset of $S^*(k)$

The rest of the paper is organized as follows. Section II formulates the problem. Section III presents the theoretical foundations for identifying node failures, followed by verifiable identifiability conditions for specific classes of probing mechanisms in Section IV. Based on the derived conditions, tight bounds on the maximum identifiability index are presented in Section V, and inner/outer bounds on the maximum identifiable set are established in Section VI. We evaluate the established bounds on various synthetic/real topologies in Section VII to study the impact of various parameters (topology, number of monitors, probing mechanism) on the capability of node failure localization. Finally, Section VIII concludes the paper.

II. PROBLEM FORMULATION

A. Models and Assumptions

We assume that the network topology is known and model it as an undirected graph[2] $\mathcal{G} = (V, L)$, where V and L are the

[2]We use the terms *network* and *graph* interchangeably.

sets of nodes and links. In $\mathcal{G}$, the number of neighbors of node v is called the *degree* of v; $\xi := |L|$ denotes the number of links. Note that graph $\mathcal{G}$ can represent a logical topology where each node in $\mathcal{G}$ corresponds to a physical subnetwork. Without loss of generality, we assume $\mathcal{G}$ is connected, as different connected components have to be monitored separately.

A subset of nodes M ($M \subseteq V$) are *monitors* that can initiate and collect measurements. The rest of the nodes, denoted by $N := V \setminus M$, are *non-monitors*. Let $\mu := |M|$ and $\sigma := |N|$ denote the numbers of monitors and non-monitors. We assume that monitors do not fail during the measurement process, as failed monitors can be directly detected and excluded (assuming centralized control within the monitoring system). Non-monitors, on the other hand, can fail, and a failure event may involve simultaneous failures of multiple non-monitors. Depending on the adopted probing mechanism, monitors measure the states of nodes by sending probes along certain paths. Let P denote the set of all *possible measurement paths*; for given $\mathcal{G}$ and M, different probing mechanisms can lead to different sets of measurement paths, which will be specified later. We use *node state* (*path state*) to refer to the binary state, failed or normal, of a node (path), where a path fails if and only if at least one node on the path fails. Table I summarizes graph-related notations used in this paper.

Let $\mathbf{w} = (W_1, \ldots, W_\sigma)^T$ be the binary column vector of the states of all non-monitors and $\mathbf{c} = (C_1, \ldots, C_\gamma)^T$ the binary column vector ($\gamma = |P|$) of the states of all measurement paths. For both node and path states, 0 represents "normal" and 1 represents "failed". We relate the path states to the node states through the following Boolean linear system:

$$\mathbf{R} \odot \mathbf{w} = \mathbf{c}, \tag{1}$$

where $\mathbf{R} = (R_{ij})$ is a $\gamma \times \sigma$ *measurement matrix*, with each entry $R_{ij} \in \{0, 1\}$ denoting whether non-monitor v_j is present on path $\mathcal{P}_i$ (1: yes, 0: no), and "$\odot$" is the Boolean matrix product, i.e., $C_i = \vee_{j=1}^{\sigma}(R_{ij} \wedge W_j)$. The goal of Boolean network tomography is to invert this Boolean linear system to solve for all/part of the elements in $\mathbf{w}$ given $\mathbf{R}$ and $\mathbf{c}$. Intuitively, for a node set S ($S \subseteq N$), any node failures in S are identifiable if and only if the corresponding states of S in $\mathbf{w}$ are uniquely determinable by (1).

B. Definitions

Let a *failure set* F be a set of non-monitors ($F \subseteq N$) that fail simultaneously. Note that the collection of all failure sets in a given network covers all possible failure scenarios (each corresponds to a failure set) that can occur in this network; the goal of failure localization is to infer the current failure set from the states of measurement paths. The challenge for this problem is that there may exist multiple failure sets leading to the same path states, causing ambiguity. Let P_F denote the set of all measurement paths affected by a failure set F (i.e., paths traversing at least one node in F). To quantify the capability of uniquely determining the failure set, we introduce the following definitions.

Definition 1. *Given a network $\mathcal{G}$ and a set of measurement paths P, two failure sets F_1 and F_2 are* distinguishable *if and only if $P_{F_1} \neq P_{F_2}$, i.e., $\exists$ a path that traverses one and only one of F_1 and F_2.*

Definition 1 implies that two potential failure sets must be associated with different observable path states for monitors to determine which set of nodes have failed. While uniquely localizing arbitrary failures requires all subsets of N to be pairwise distinguishable, we can relax this requirement by only considering failure sets of size bounded by k ($k \geq 1$), which represents the scale of probable failure events. Moreover, in practice, we are usually interested in the states of a subset of nodes S ($S \subseteq N$), in which case the goal is to only ensure unique failure localization within S. Note that failures (F) may occur anywhere in the network ($F \subseteq N$) and are not restricted to S.

Definition 2. *Given a network $\mathcal{G}$ (with non-monitor set N) and a node set S of interest ($S \subseteq N$):*

1) *S is* k-identifiable *if for any two failure sets F_1 and F_2 satisfying (1) $|F_i| \leq k$ ($i = 1, 2$) and (2) $F_1 \cap S \neq F_2 \cap S$, F_1 and F_2 are distinguishable.*
2) *The* maximum identifiability index *of S, denoted by $\Omega(S)$, is the maximum value of k such that S is k-identifiable.*

Intuitively, if a node set S is k-identifiable, then the states (normal/failed) of all nodes within this set are unambiguously determinable from the observed path states, provided the total number of failures (anywhere in the network) is bounded by k. The maximum identifiability index $\Omega(S)$ characterizes the network's capability to uniquely localize failures in S. Definition 2 generalizes the notion of network-wide k-identifiability and maximum identifiability index introduced in [17], where only the case of $S = N$ was considered. In the special case of $S = \{v\}$, we say that node v is k-identifiable; the maximum identifiability index of $S = \{v\}$ is denoted by $\Omega(v)$. Note that the subset of a k-identifiable set is also k-identifiable. We are therefore interested in the maximum such set.

Definition 3. *Given k, the maximum k-identifiable set, denoted by $S^*(k)$, is the largest-cardinality non-monitor set that is k-identifiable.*

According to Definition 3, it seems that the maximum k-identifiable set is defined based on its cardinality, and thus may not be unique. Nevertheless, we prove in Section III-B that $S^*(k)$ is unique. The significance of the maximum k-identifiable set is that it measures the completeness of the inferred network state: it contains all nodes whose states can be inferred reliably from the observed path states, as long as the total number of failures in the network is bounded by k. Note that k is a design parameter capturing the scale of failures that the system is designed to handle.

C. Classification of Probing Mechanisms

The above definitions are all defined with respect to (w.r.t.) a given set of measurement paths P. Given the topology $\mathcal{G}$ and monitor locations M, the probing mechanism plays a crucial role in determining P. Depending on the flexibility of probing

and the cost of deployment, we classify probing mechanisms into one of three classes:

1) *Controllable Arbitrary-path Probing (CAP):* P includes any path/cycle, allowing repeated nodes/links, provided each path/cycle starts and ends at monitors.
2) *Controllable Simple-path Probing (CSP):* P includes any *simple* path between distinct monitors, not including repeated nodes.
3) *Uncontrollable Probing (UP):* P is the set of paths between monitors determined by the routing protocol used by the network, not controllable by the monitors.

Although CAP allows probes to traverse each node/link an arbitrary number of times, it suffices to consider paths where each probe traverses each link at most once in either direction for the sake of localizing node failures.

These probing mechanisms clearly provide decreasing flexibility to the monitors and therefore decreasing capability to localize failures. However, they also offer decreasing deployment cost. CAP represents the most flexible probing mechanism and provides an upper bound on failure localization capability. In traditional networks, CAP is feasible at the IP layer if *(strict) source routing* (an IP option) [19] is enabled at all nodes[3], or at the application layer (to localize failures in overlay networks) if equivalent "source routing" is supported by the application. Moreover, CAP is also feasible under an emerging networking paradigm called software-defined networking (SDN) [20], [21], where monitors can instruct the SDN controller to set up arbitrary paths for the probing traffic. In particular, an SDN consisting of OpenFlow switches [21] can set up paths by configuring the flow table of each traversed OpenFlow switch to forward a probing flow (e.g., one TCP connection) to a next hop based on the ingress port and the flow identifier, which allows the path to have repeated nodes/links. In contrast, UP represents the most basic probing mechanism, feasible in any network supporting data forwarding, that provides a lower bound on the capability of failure localization. CSP represents an intermediate case that allows control over routing while respecting a basic requirement that routes must be cycle-free. CSP is implementable by MPLS (MultiProtocol Label Switching), where the "explicit routing" mode [22] allows one to set up a controllable, non-shortest path using labels so long as the path are cycle-free. Note that the cycle-free constraint here is crucial, as data forwarding in MPLS will encounter forwarding loops if a path has cycles.

The significance of these three probing mechanisms is that they capture the main features of several existing and emerging routing techniques. Specifically, UP is generally supported in existing networks without special configuration, CSP is feasible in some of today's networks running MPLS with certain configuration (i.e., label propagation via explicit routing), while CAP represents the capability of future networks once SDN is broadly deployed.

[3]Source routing allows nodes to modify the source and the destination addresses in packet headers hop by hop along the path prescribed by a monitor. The probe can follow the reverse path to return to the original monitor, thus effectively probing any path with *at least* one end at a monitor.

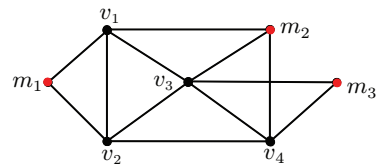


Fig. 1. Sample network with three monitors: m_1, m_2, and m_3.

Discussion: In [23], "m-trail" (monitoring trails) is employed as a probing mechanism in all-optical networks, where measurement paths can contain repeated nodes but *not* repeated links. It is unclear which routing protocols in communication networks select paths under the restriction of "m-trails", we thus do not consider such a probing mechanism in this paper. In [16], another probing mechanism "m-tour" (monitoring tours) is used, which allows both repeated nodes and repeated links in measurement paths; "m-tour" is equivalent to CAP.

In this paper, we quantify how the flexibility of a probing scheme affects the network's capability to localize failures. Although concrete results are only provided for the above classes of probing mechanisms, our framework and our abstract identifiability conditions (see Section III-A) can also be used to evaluate the failure localization capabilities of other probing mechanisms.

D. Objective

Given a network topology $\mathcal{G}$, a set of monitors M, and a probing mechanism (CAP, CSP, or UP), we seek to answer the following closely related questions: (i) Given a node set of interest S and a bound k on the number of failures, can we uniquely localize up to k failed nodes in S from observed path states? (ii) Given a node set S, what is the maximum number of failures within S that can be uniquely localized? (iii) Given an integer k ($1 \leq k \leq \sigma$), what is the largest node set that is k-identifiable? We will study these problems from the perspectives of both theories and efficient algorithms.

E. Illustrative Example

Consider the sample network in Fig. 1 with three monitors (m_1–m_3) and four non-monitors (v_1–v_4). Under UP, suppose that the default routing protocol only allows the monitors to probe the following paths: $\mathcal{P}_1 = m_1v_1m_2$, $\mathcal{P}_2 = m_2v_4m_3$, and $\mathcal{P}_3 = m_1v_2v_4m_3$, which form a measurement matrix $\mathbf{R}^{\text{UP}}$:

$$\begin{array}{l} \mathcal{P}_1 = m_1v_1m_2 \\ \mathcal{P}_2 = m_2v_4m_3 \\ \mathcal{P}_3 = m_1v_2v_4m_3 \end{array} \Rightarrow \mathbf{R}^{\text{UP}} = \begin{array}{c} \begin{array}{cccc} W_1 & W_2 & W_3 & W_4 \end{array} \\ \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 1 \end{pmatrix} \end{array}, \tag{2}$$

where $R^{\text{UP}}_{ij} = 1$ if and only if node v_j is on path $\mathcal{P}_i$. Then we have $\mathbf{R}^{\text{UP}} \odot \mathbf{w} = \mathbf{c}$, where $\mathbf{c}$ is the binary vector of path states observed at the destination monitors. Let $S' := \{v_1, v_2, v_4\}$. Based on Definition 3, we can verify that $\Omega(S') = 2$, and the maximum identifiable set $S^*(1) = \{v_1, v_2, v_4\}$ and $S^*(2) = S^*(3) = S^*(4) = \{v_1, v_4\}$. Under CSP, besides the three paths in (2), we can probe three additional paths: $\mathcal{P}_4 = m_2v_3m_3$, $\mathcal{P}_5 = m_1v_2v_3m_3$, and $\mathcal{P}_6 = m_1v_2v_1m_2$, yielding an expanded measurement matrix in (3):

$$\begin{array}{l}\mathcal{P}_1 = m_1 v_1 m_2 \\ \mathcal{P}_2 = m_2 v_4 m_3 \\ \mathcal{P}_3 = m_1 v_2 v_4 m_3 \\ \mathcal{P}_4 = m_2 v_3 m_3 \\ \mathcal{P}_5 = m_1 v_2 v_3 m_3 \\ \mathcal{P}_6 = m_1 v_2 v_1 m_2\end{array} \Rightarrow \mathbf{R}^{\text{CSP}} = \begin{array}{c} \begin{array}{cccc} W_1 & W_2 & W_3 & W_4 \end{array} \\ \left(\begin{array}{cccc} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 1 \\ \hline 0 & 0 & 1 & 0 \\ 0 & 1 & 1 & 0 \\ 1 & 1 & 0 & 0 \end{array}\right) \end{array} \left.\begin{array}{c} \\ \\ \\ \\ \\ \\ \end{array}\right\} \mathbf{R}^{\text{UP}} \tag{3}$$

Using the six paths in (3), the maximum identifiability index of S' becomes $\Omega(S') = 3$, and the maximum identifiable set is enlarged to $S^*(1) = S^*(2) = S^*(3) = \{v_1, v_2, v_3, v_4\}$ and $S^*(4) = \{v_1, v_3, v_4\}$, a notable improvement over UP. Finally, if CAP is supported, then we can send probes along a cycle $\mathcal{P}_7 = m_1 v_2 m_1$. In conjunction with the paths in (3), this yields the measurement matrix in (4):

$$\begin{array}{l}\mathcal{P}_1 = m_1 v_1 m_2 \\ \mathcal{P}_7 = m_1 v_2 m_1 \\ \mathcal{P}_4 = m_2 v_3 m_3 \\ \mathcal{P}_2 = m_2 v_4 m_3\end{array} \Rightarrow \mathbf{R}^{\text{CAP}} = \begin{array}{c} \begin{array}{cccc} W_1 & W_2 & W_3 & W_4 \end{array} \\ \left(\begin{array}{cccc} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{array}\right) \end{array} \tag{4}$$

Since the paths in (4) can independently determine the state of each non-monitor, we have $\Omega(S') = 4$ and $S^*(1) = S^*(2) = S^*(3) = S^*(4) = \{v_1, v_2, v_3, v_4\}$ under CAP, i.e., all failures can be uniquely localized.

This example shows that the monitor placement and the probing mechanism significantly affect a network's capability to localize failures. In the rest of the paper, we will study this relationship both theoretically and algorithmically.

III. Theoretical Foundations

We start with some basic understanding of failure identifiability. First, the definition of k-identifiability in Definition 2 requires enumeration of all possible failure events and thus cannot be tested efficiently. To address this issue, we establish explicit sufficient/necessary conditions for k-identifiability that apply to arbitrary probing mechanisms, which will later be developed into verifiable conditions for the three classes of probing mechanisms. Moreover, we establish several desirable properties of maximum identifiability index (Definition 2) and maximum identifiable set (Definition 3), which greatly simplify the computation of these measures.

A. Abstract Identifiability Conditions

Our identifiability condition is inspired by a result known in a related field called *combinatorial group testing* [24]. In short, group testing aims to find abnormal elements in a given set by running tests on subsets of elements, each test indicating whether any element in the subset is abnormal. This is analogous to our problem where abnormal elements are failed nodes and tests are conducted by probing measurement paths. A subtle but critical difference is that in our problem, the subsets of elements that can be tested together are constrained by the set of measurement paths P, which is in turn limited by the topology, probing mechanism, and placement of monitors[4].

[4]In this regard, our problem is similar to a variation of group testing under graph constraints [18]; see Section I-A for the difference.

Most existing solutions for (nonadaptive) group testing aim at constructing a *disjunct testing matrix*. Specifically, a testing matrix R is a binary matrix, where $R_{i,j} = 1$ if and only if element j is included in the i-th test. Matrix R is k-*disjunct* if the Boolean sum of any k columns does not "contain" any other column[5] [25]. In our problem, the existence of a disjunct testing matrix translates into the following conditions.

Lemma 4. *Set S is k-identifiable:*

a) if for any failure set F with $|F| \leq k$ and any node v with $v \in S \setminus F$, $\exists\ p \in P$ traversing v but none of the nodes in F;

b) only if for any failure set F with $|F| \leq k-1$ and any node v with $v \in S \setminus F$, $\exists\ p \in P$ traversing v but none of the nodes in F.

Proof: Consider two distinct failure sets F_1 and F_2 with $F_1 \cap S \neq F_2 \cap S$, each containing no more than k nodes. There exists a node $v \in S$ in only one of these sets; suppose $v \in F_1 \setminus F_2$. By the condition in the lemma, $\exists$ a path p traversing v but not F_2, thus distinguishing F_1 from F_2. Therefore, condition *a)* in Lemma 4 is sufficient.

Suppose $\exists$ a non-empty set F with $|F| \leq k-1$ and $v \in S \setminus F$ such that all measurement paths traversing v must also traverse at least one node in F. Therefore, for two failure sets F and $F \cup \{v\}$ satisfying conditions (1–2) in Definition 2-(1) are not distinguishable as $P_F = P_{F \cup \{v\}}$. Thus, condition *b)* in Lemma 4 is necessary. ■

These conditions generally apply to any probing mechanism. Although in the current form, they do not directly lead to efficient testing algorithms, we will show later (Section IV) that they can be transformed into verifiable conditions for several classes of probing mechanisms.

B. Properties of the Maximum Identifiability Index and the Maximum Identifiable Set

Although the maximum identifiability index $\Omega(S)$ and the maximum k-identifiable set $S^*(k)$ are defined for sets of nodes, we show below that they can both be characterized in terms of a per-node property, which greatly simplifies the computation of these measures. We start with the following two observations.

Lemma 5. *a) If S is k-identifiable, then any $v \in S$ must be k-identifiable.*

b) If v is k-identifiable $\forall v \in S$, then S is k-identifiable.

Proof: a) Suppose $\exists$ node $v \in S$ that is not k-identifiable, then $\exists$ at least two failure sets F_1 and F_2 with $|F_i| \leq k$ ($i = \{1, 2\}$) and $F_1 \cap \{v\} \neq F_2 \cap \{v\}$ such that F_1 and F_2 are not distinguishable. Thus, S is not k-identifiable as $v \in S$.

b) For any two failure sets F_1 and F_2 with $|F_i| \leq k$ ($i = \{1, 2\}$) and $F_1 \cap S \neq F_2 \cap S$, $\exists$ a node $v \in S$ that is either in F_1 or F_2 but not both. Since node v is k-identifiable, F_1 and F_2 must be distinguishable. Therefore, S is k-identifiable. ■

Proposition 6. $\Omega(S) = \min_{v \in S} \Omega(v)$.

[5]That is, for any subset of k column indices S and any other column index $j \notin S$, $\exists$a row index i such that $R_{i,j} = 1$ and $R_{i,j'} = 0$ for all $j' \in S$.

Proof: By Lemma 5-(a), any $v \in S$ must have $\Omega(v) \geq \Omega(S)$. Thus, $\min_{v\in S} \Omega(v) \geq \Omega(S)$. By the definition of maximum identifiability index, all nodes in S are $\min_{v\in S} \Omega(v)$-identifiable. By Lemma 5-(b), S is also $\min_{v\in S} \Omega(v)$-identifiable. Thus, $\Omega(S) \geq \min_{v\in S} \Omega(v)$. Therefore, $\Omega(S) = \min_{v\in S} \Omega(v)$. ■

Corollary 7. *Maximum identifiability index of S, $\Omega(S)$, is monotonically non-increasing in the sense that $\Omega(S_1) \geq \Omega(S_2)$ for any two non-empty sets S_1 and S_2 with $S_1 \subset S_2$.*

Proof: Since $S_1 \subset S_2$, $\min_{v\in S_1} \Omega(v) \geq \min_{v\in S_2} \Omega(v)$. Therefore, by Proposition 6, $\Omega(S_1) \geq \Omega(S_2)$. ■

Therefore, we can estimate the maximum identifiability index of a given non-monitor set using Corollary 7 when the maximum identifiability index of its subset/superset is known.

Next, we show that maximum k-identifiable sets exhibit properties that can facilitate fast determination of which nodes should be included/excluded in these sets.

Proposition 8. *Let $S'(k) := \{v \in N : v$ is k-identifiable $\}$. Then $S'(k) = S^*(k)$.*

Proof: By Lemma 5-(a), any node in $S^*(k)$ is k-identifiable. Therefore, $S^*(k) \subseteq S'(k)$.

Next, $S'(k)$ must be k-identifiable according to Lemma 5-(b). Thus $|S'(k)| \leq |S^*(k)|$. Consequently, $S'(k) = S^*(k)$. ■

Proposition 8 provides a method to construct the maximum k-identifiable set $S^*(k)$ by simply collecting all k-identifiable nodes. Based on this method, we can further prove the uniqueness and monotonicity of $S^*(k)$ as follows:

Corollary 9. *The maximum k-identifiable set $S^*(k)$ is unique and monotonically non-increasing in k, i.e., $S^*(k+1) \subseteq S^*(k)$ for any k.*

Proof: Definition 2 implies that k-identifiability is a per-node property that is independent of the identifiability of other nodes. Therefore, for each node in N, it is either k-identifiable or *not* k-identifiable. By Proposition 8, $S^*(k)$ is a set containing all k-identifiable nodes; therefore, $S^*(k)$ is unique.

For each node $w \in N \setminus S^*(k)$, w is *not* k-identifiable, and thus w is not $(k+1)$-identifiable. Since $S^*(k+1)$ is a collection of all $(k+1)$-identifiable nodes, no nodes in $N \setminus S^*(k)$ can be included in $S^*(k+1)$. Thus, $S^*(k+1) \subseteq S^*(k)$. ■

Intuitively, if there exists a k-identifiable set $S'(k)$ with $|S'(k)| = |S^*(k)|$, then we must have $S'(k) = S^*(k)$. Thus, Corollary 9 suggests one way to obtain $S^*(k)$ is to identify $S^*(j)$ for $j < k$ and then only study subsets of $S^*(j)$; nodes outside $S^*(j)$ are guaranteed to be excluded from $S^*(k)$.

Corollary 10. *Let $S''(k) := \{v \in N : \exists$ path in P traversing v but none of the nodes in each failure set F with $v \notin F$ and $|F| \leq k\}$. Then $S''(k) \subseteq S^*(k)$.*

Proof: $S''(k)$ satisfies sufficient condition a) in Lemma 4. Thus, $\Omega\big(S''(k)\big) \geq k$. Following similar arguments as in the proof of Proposition 8, again we have that each node in $S''(k)$ is at least k-identifiable. Therefore, $S''(k) \subseteq S^*(k)$. ■

By Corollary 10, we note that $S''(k)$ underestimates the size of the maximum k-identifiable set $S^*(k)$, yet it forms an inner bound (i.e., subset) of $S^*(k)$, thus providing theoretical support for determining the must-have nodes in the optimum set $S^*(k)$; see detailed discussions presented in Section VI.

Remark: Results in this section apply to any probing mechanism. We will show in the following sections how they can be used to design efficient algorithms for probing mechanisms CAP, CSP, and UP. The above results can also be used to design algorithms for other probing mechanisms.

IV. Verifiable Identifiability Conditions

In this section, starting from the abstract conditions in Section III-A, we develop concrete conditions suitable for efficient testing for the three classes of probing mechanisms.

A. Conditions under CAP

CAP essentially allows us to "ping" any node from a monitor along any path. In the face of failures, this allows a monitor to determine the state of a node as long as it is connected to the node after removing other failed nodes. This observation allows us to translate the conditions in Section III-A into more concrete identifiability conditions (Lemma 11).

Lemma 11. *Set S is k-identifiable under CAP if and only if for any set V' of up to $k-1$ non-monitors, each connected component in $\mathcal{G} - V'$ that contains a node in S has a monitor.*

Proof: Necessity. Suppose the above condition does not hold, i.e., there exists a non-monitor v ($v \in S$) that is disconnected from all monitors in $\mathcal{G} - V'$ for a set V' of up to $k-1$ non-monitors ($v \notin V'$). Then if nodes in V' fail, no remaining measurement path can probe v, violating the condition in Lemma 4-(b).

Sufficiency. The proof is similar to that of Theorem 2 in [16], except that we are only interested in localizing failures in S. Consider two failure sets F_1 and F_2 with $|F_i| \leq k$ ($i = \{1, 2\}$) and $F_1 \cap S \neq F_2 \cap S$. Then $\exists$ node v ($v \in S$) that is in one and only one of F_1 and F_2. Without loss of generality, let $v \in F_1$. Let $I := F_1 \cap F_2$. Since $|I| \leq k-1$, $\exists$ a path p connecting a monitor m with node v in $\mathcal{G} - I$ if the condition in Lemma 11 holds. Let w be the first node on p (starting from m) that is in either $F_1 \setminus I$ or $F_2 \setminus I$. Truncating p at w gives a path p' such that p' and its reverse path form a measurement path from m to w and back to m that traverses only F_1 or F_2, thus distinguishing F_1 and F_2. ■

Under CAP, Lemma 11 shows that the necessary condition derived from Lemma 4 is also sufficient. However, the condition in Lemma 11 still cannot be tested efficiently because a combinatorial number of sets V' are enumerated. Fortunately, we can reduce Lemma 11 into explicit conditions on vertex-cuts of a related topology, which can then be tested in polynomial time. We use the following notion from graph theory.

Definition 12. *For two nodes s and t in an undirected graph $\mathcal{G}$, (s,t)-vertex-cut in $\mathcal{G}$, denoted by $C_{\mathcal{G}}(s,t)$, is the minimum-cardinality node set whose deletion destroys all paths from s to t. If s and t are neighbors, $C_{\mathcal{G}}(s,t) := V(\mathcal{G}) \setminus \{t\}$.*

Our key observation is that requiring each connected component in $\mathcal{G} - V'$ that contains a node in S to have a

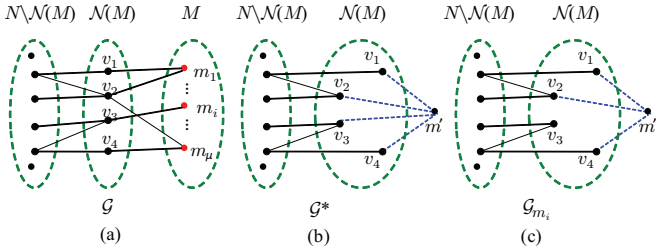


Fig. 2. Auxiliary graphs: (a) Original graph $\mathcal{G}$; (b) $\mathcal{G}^*$ of $\mathcal{G}$; (c) $\mathcal{G}_{m_i}$ of $\mathcal{G}$ w.r.t. monitor m_i.

monitor is equivalent to requiring each such component in $\mathcal{G} - M - V'$ (i.e., after removing all monitors) to contain a neighbor of a monitor. Thus, if we extend $\mathcal{G} - M$ by adding a *virtual monitor* m' and *virtual links* connecting m' and all neighbors of monitors to obtain an *auxiliary graph* $\mathcal{G}^* := \mathcal{G} - M + \{m'\} + \mathcal{L}\big(\{m'\}, \mathcal{N}(M)\big)$ (illustrated in Fig. 2 (b)), then each node in $S \setminus V'$ should be connected to m' in $\mathcal{G}^* - V'$. In other words, the minimum cardinality of the (m', w)-vertex-cut in $\mathcal{G}^*$ over all $w \in S$ must be greater than $|V'|$. For ease of presentation, we introduce the following definition.

Definition 13. *Given a graph $\mathcal{G}$, a node set S, and a node $m \notin S$, define $\Gamma_{\mathcal{G}}(S, m) := \min_{w \in S} |C_{\mathcal{G}}(w, m)|$.*

By this definition, Lemma 11 can be transformed into a new condition, which reduces the tests over all possible V' to a single test of the vertex-cuts of $\mathcal{G}^*$, as stated below (recall that σ is the total number of non-monitors).

Lemma 14. *Each connected component in $\mathcal{G} - V'$ that contains a node in S has a monitor for any set V' of up to q ($q \leq \sigma - 1$) non-monitors if and only if $\Gamma_{\mathcal{G}^*}(S, m') \geq q + 1$.*

Proof: The proof can be found in [26]. ■

Lemma 14 allows us to rewrite the identifiability conditions in Lemma 11 in terms of the vertex-cuts of $\mathcal{G}^*$.

Theorem 15 (k-identifiability under CAP)**.** *Set S is k-identifiable ($k \leq \sigma$) under CAP if and only if $\Gamma_{\mathcal{G}^*}(S, m') \geq k$.*

A special case of Theorem 15 occurs when $k = \sigma$, i.e., any non-monitors can fail simultaneously. In this case, each node in S must directly connect to at least one monitor in $\mathcal{G}$.

Discussion: Theorem 15 extends and improves the identifiability condition given in Theorem 2 of [16] by (i) considering failures within an arbitrary subset of nodes instead of the entire network, and (ii) providing a single condition that can be tested in polynomial time (see testing algorithm below) instead of testing a combinatorial number of conditions that enumerate all possible failure events.

Testing algorithm: A key advantage of the newly derived conditions over the abstract conditions in Section III-A is that they can be tested efficiently. Let $\theta := |\mathcal{N}(M)|$ denote the number of non-monitors that are neighbors of at least one monitor in M. Given node w, $C_{\mathcal{G}^*}(w, m')$ can be computed in $O(\theta\xi)$ time[6], where ξ is the number of links (refer to Table I for notations). Therefore, we can evaluate $\Gamma_{\mathcal{G}^*}(S, m')$ in $O(\theta\xi|S|)$ time and compare the result with k to test the conditions in Theorem 15.

[6]The (m', w)-vertex-cut problem in an undirected graph can be reduced to an (m', w)-edge-cut problem in a directed graph in linear time [27]. The (m', w)-edge-cut problem is solvable by the Ford−Fulkerson algorithm [28] in $O(\theta\xi)$ time.

B. Conditions under CSP

Under CSP, we restrict measurement paths P to the set of *simple paths* between monitors, i.e., paths starting/ending at distinct monitors that contain no cycles. As in CAP, our goal is again to transform the abstract conditions in Section III-A into concrete sufficient/necessary conditions that can be efficiently verified. We first give analogous result to Theorem 15.

Lemma 16. *Set S is k-identifiable under CSP:*

a) if for any node set V', $|V'| \leq k + 1$, containing at most one monitor, each connected component in $\mathcal{G} - V'$ that contains a node in S also contains a monitor;

b) only if for any node set V', $|V'| \leq k$, containing at most one monitor, each connected component in $\mathcal{G} - V'$ that contains a node in S also contains a monitor.

Proof: The proof can be found in [26]. ■

Due to the restriction to simple paths, the identifiability conditions in Lemma 16 are stronger than those in Lemma 11. As with Lemma 11, the conditions in Lemma 16 do not directly lead to efficient tests, and we again seek equivalent conditions in terms of topological properties. Each condition in the form of Lemma 16 (a–b) covers two cases: (i) V' only contains non-monitors; (ii) V' contains a monitor and $|V'|-1$ non-monitors. The first case has been converted to a vertex-cut property on an auxiliary topology $\mathcal{G}^*$ by Lemma 14; we now establish a similar condition for the second case using similar arguments.

Fix a set $V' = F \cup \{m\}$, where m is a monitor in M and F a set of non-monitors. Again, the key observation is that each connected component in $\mathcal{G} - V'$ that contains a node in S also containing a monitor is equivalent to each such component in $\mathcal{G} - M - F$ containing a neighbor of a monitor other than m (i.e., a node in $\mathcal{N}(M \setminus \{m\})$). To capture this observation, we introduce another *auxiliary graph* $\mathcal{G}_m := \mathcal{G} - M + \{m'\} + \mathcal{L}\big(\{m'\}, \mathcal{N}(M \setminus \{m\})\big)$ w.r.t. monitor m as illustrated in Fig. 2 (c), where m' is again a virtual monitor. We will show that the second case (V' contains a monitor) is equivalent to requiring that the nodes in $S \setminus F$ and m' are in the same connected component within $\mathcal{G}_m - F$, and thus the following holds.

Lemma 17. *The following two conditions are equivalent:*

(1) Each connected component in $\mathcal{G} - V'$ that contains a node in S also contains a monitor for $\forall$sets V' containing monitor m ($m \in M$) and up to q ($q \leq \sigma - 1$) non-monitors;

(2) $\Gamma_{\mathcal{G}_m}(S, m') \geq q + 1$.

Proof: The proof can be found in [26]. ■

Based on Lemmas 14 and 17, we can rewrite Lemma 16 as follows.

Theorem 18 (k-identifiability under CSP)**.** *Set S is k-identifiable under CSP:*

a) if $\Gamma_{\mathcal{G}^*}(S, m') \geq k+2$, and $\min_{m \in M} \Gamma_{\mathcal{G}_m}(S, m') \geq k+1$ ($k \leq \sigma - 2$);

b) only if $\Gamma_{\mathcal{G}^*}(S, m') \geq k+1$*, and* $\min_{m \in M} \Gamma_{\mathcal{G}_m}(S, m') \geq k$ *(*$k \leq \sigma - 1$*).*

Theorem 18 does not include the cases of $k = \sigma$ and $k = \sigma - 1$, which are addressed in Propositions 19 and 20.

Proposition 19. *Set* S *is* σ*-identifiable under CSP if and only if each node in* S *has at least two monitors as neighbors.*

Proof: The proof can be found in [26]. ■

Proposition 20. *Set* S *is* $(\sigma - 1)$*-identifiable under CSP if and only if (i) all nodes in* S *have at least two monitors as neighbors, or (ii) all nodes in* $N \setminus \{v\}$ *(*$v \in S$*) have at least two monitors as neighbors and* v *has all nodes in* $N \setminus \{v\}$ *and one monitor as neighbors.*

Proof: The proof can be found in [26]. ■

Testing algorithm: Similar to the case of CAP, we can use the algorithm in [27], [28] to compute the vertex-cuts of the auxiliary graphs $\mathcal{G}^*$ and $\mathcal{G}_m$ ($\forall m \in M$), and test the conditions in Theorem 18 for any given k. The overall complexity of the test is $O(\mu\theta\xi|S|)$ (refer to Table I for notations).

C. Conditions under UP

Under UP, monitors have no control over the probing paths between monitors, and the set of measurement paths P is limited to the paths between monitors determined by the network's native routing protocol. In contrast to the previous cases (CAP, CSP), identifiability under UP can no longer be characterized in terms of topological properties. We can, nevertheless, establish explicit conditions based on the abstract conditions in Section III-A. The idea is to examine how many non-monitors need to be removed to disconnect all measurement paths traversing a given non-monitor v. If the number is sufficiently large (greater than k), then we can still infer the state of v from some measurement path when a set of other non-monitors fail; if the number is too small (smaller than or equal to $k-1$), then we are not able to determine the state of v as the failures of all paths traversing v can already be explained by the failures of other non-monitors. This intuition leads to the following results.

In the sequel, $P_v \subseteq P$ denotes the set of measurement paths traversing a non-monitor v, and $\mathcal{C}_v := \{P_w : w \in N, w \neq v\}$ denotes the collection of path sets traversing non-monitors in $N \setminus \{v\}$. We use $\text{MSC}(v)$ to denote the size of the *minimum set cover* of P_v by $\mathcal{C}_v$, i.e., $\text{MSC}(v) := |V'|$ for the minimum set $V' \subseteq N \setminus \{v\}$ such that $P_v \subseteq \bigcup_{w \in V'} P_w$. Note that covering is only feasible if v is not on any 2-hop measurement path (i.e., monitor-v-monitor), in which case we know $P_v \subseteq \bigcup_{w \in N, w \neq v} P_w$ and thus $\text{MSC}(v) \leq \sigma - 1$. If v is on a 2-hop path, then we define $\text{MSC}(v) := \sigma$.

Theorem 21 (k-identifiability under UP)**.** *Set* S *is* k*-identifiable under UP with measurement paths* P*:*

a) if $MSC(v) \geq k+1$ *for any node* v *in* S *(*$k \leq \sigma - 1$*);*
b) only if $MSC(v) \geq k$ *for any node* v *in* S *(*$k \leq \sigma$*).*

Proof: The proof can be found in [26]. ■

The only case not considered by Theorem 21 is the case that $k = \sigma$, for which we develop the following condition.

Proposition 22. *Set* S *is* σ*-identifiable under UP if and only if* $MSC(v) = \sigma$ *for any node* v *in* S*, i.e., each node in* S *is on a 2-hop path.*

Proof: The proof can be found in [26]. ■

Testing algorithm: The conditions in Theorem 21 provide an explicit way to test k-identifiability under UP, using tests of the form $\text{MSC}(v) \geq q$. Unfortunately, evaluating such a test, known as the decision problem of the *set covering problem*, is known to be NP-complete. Nevertheless, we can use approximation algorithms to compute bounds on $\text{MSC}(v)$. An algorithm with the best approximation guarantee is the *greedy algorithm*, which iteratively selects the set in $\mathcal{C}_v$ that contains the largest number of uncovered paths in P_v until all the paths in P_v are covered (assuming that v is not on any 2-hop path).

Let $\text{GSC}(v)$ denote the number of sets selected by the greedy algorithm. This immediately provides an upper bound: $\text{MSC}(v) \leq \text{GSC}(v)$. Moreover, since the greedy algorithm has an approximation ratio of $\log(|P_v|)+1$ [29], we can also bound $\text{MSC}(v)$ from below: $\text{MSC}(v) \geq \text{GSC}(v)/(\log(|P_v|)+1)$. Applying these bounds to Theorem 21 yields relaxed conditions:

- S is k-identifiable under UP if $k < \lceil \min_{v \in S} \frac{\text{GSC}(v)}{\log(|P_v|)+1} \rceil$;
- S is *not* k-identifiable under UP if $k > \min_{v \in S} \text{GSC}(v)$.

These conditions can be tested by running the greedy algorithm for all nodes in S, each taking time $O(|P_v|^2\sigma) = O(|P|^2\sigma)$, and the overall test has a complexity of $O(|S||P|^2\sigma)$ (or $O(\mu^4\sigma|S|)$ if there is a measurement path between each pair of monitors).

D. Special Case: 1-identifiability

In practice, the most common failure event consists of the failure of a single node. Thus, an interesting question is whether S is 1-identifiable under a given monitor placement and a given probing mechanism. In our previous results, Theorems 18 and 21 only provide an answer to the above question if the sufficient condition is satisfied or the necessary condition is violated for $k = 1$; however, the answer is unknown if S satisfies the necessary condition but violates the sufficient condition under CSP and UP. In contrast, Theorem 15 establishes a condition under CAP that is both necessary and sufficient, yet still expressed in a complicated form (i.e., vertex-cuts). We develop explicit methods below for testing S for 1-identifiability.

1) Conditions for 1*-identifiability:* We start with a generic necessary and sufficient condition that applies to all probing mechanisms. Recall that P_v denotes the set of measurement paths traversing a non-monitor v. For $k = 1$, Definition 2-(1) is equivalent to the following:

Claim 23. S *is* 1*-identifiable if and only if:*

(1) $P_v \neq \emptyset$ *for any* $v \in S$*, and*
(2) $P_v \neq P_w$ *for any* $v \in S$*,* $w \in N$*, and* $v \neq w$*.*

In Claim 23, the first condition guarantees that any failure in S is detectable (i.e., causing at least one path failure), and the second condition guarantees that the observed path states can uniquely localize the failed node in S. An efficient test of these conditions, however, requires different strategies for different probing mechanisms.

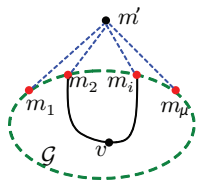


Fig. 3. Extended graph $\mathcal{G}'$.

2) Test under CAP: By Theorem 15, S is 1-identifiable under CAP if and only if $\Gamma_{\mathcal{G}^*}(S, m') \geq 1$. This is equivalent to requiring that $\mathcal{G}^*$ be connected, i.e., $\mathcal{G}$ has one monitor.

Testing for 1-identifiability of S under CAP is therefore reduced to determining if the network has a monitor.

3) Test under CSP: Under CSP, we derive conditions that are equivalent to those in Claim 23 but easier to test.

Condition (1) in Claim 23 requires that every non-monitor in S reside on a monitor-monitor simple path. While an exhaustive search for such a path incurs an exponential cost, we can test for its existence efficiently using the following observation. The idea is to construct an *extended graph* $\mathcal{G}' := \mathcal{G} + \{m'\} + \mathcal{L}(\{m'\}, M)$, i.e., by adding a virtual monitor m' and connecting it to all the monitors; see an illustration in Fig. 3. We claim that a non-monitor v is on a monitor-monitor simple path if and only if the size of the (m', v)-vertex-cut in $\mathcal{G}'$ is at least two, i.e., $\Gamma_{\mathcal{G}'}(v, m') \geq 2$, which implies the existence (see Definition 12) of two vertex-independent simple paths between v and m', illustrated as paths vm_2m' and vm_im' in Fig. 3. Truncating these two paths at m_2 and m_i yields two path segments vm_2 and vm_i, whose concatenation gives a monitor-to-monitor simple path traversing v, i.e., m_2vm_i in Fig. 3. On the other hand, if $\exists$ a monitor-to-monitor simple path traversing v, then it can be split into two simple paths connecting v to two distinct monitors, which implies $\Gamma_{\mathcal{G}'}(v, m') \geq 2$ as each of these two distinct monitors connects to m' by a virtual link.

Condition (2) in Claim 23 is violated if and only if there exist two non-monitors $v \neq w$ (at least one of them in S) such that all monitor-to-monitor simple paths traversing v must traverse w (i.e., $P_v \subseteq P_w$) and vice versa. Since $P_v \subseteq P_w$ means that there is no monitor-to-monitor simple path traversing v in $\mathcal{G} - \{w\}$, by the above argument, we see that $P_v \subseteq P_w$ if and only if the size of the (m', v)-vertex-cut in a new graph $\mathcal{G}'_w := \mathcal{G} - \{w\} + \{m'\} + \mathcal{L}(\{m'\}, M)$ is smaller than two. Therefore, condition (2) in Claim 23 is satisfied if and only if for every two distinct non-monitors v ($v \in S$) and w, either the (m', v)-vertex-cut in $\mathcal{G}'_w$ or the (m', w)-vertex-cut in $\mathcal{G}'_v$ contains two or more nodes.

In summary, the necessary and sufficient condition for 1-identifiability under CSP is:

i) $\Gamma_{\mathcal{G}'}(S, m') \geq 2$, and
ii) $\Gamma_{\mathcal{G}'_w}(v, m') \geq 2$ or $\Gamma_{\mathcal{G}'_v}(w, m') \geq 2$ for all $v \in S$, $w \in N$, and $v \neq w$.

Since $\Gamma_{\mathcal{G}}(v, w) \geq 2$ can be tested in $O(|V| + |L|)$ time[7], the overall test takes $O(\sigma|S|(|V|+|L|)) = O(\sigma(\mu+\sigma)^2|S|)$ time.

4) Test under UP: Under UP, the total number of measurement paths $|P|$ is reduced to $O(\mu^2)$ (from exponentially many as in the case of CAP/CSP) as the measurable routes are predetermined. This reduction makes it feasible to directly test conditions (1–2) in Claim 23 by testing condition (1) for each node in S and condition (2) for each pair of non-monitors (one of which is in S). Then the overall complexity of is $O(\sigma\mu^2|S|)$, dominated by testing of condition (2) in Claim 23.

[7]We can compute the biconnected component decomposition [30] and test if v and w belong to the same biconnected component.

V. Characterization of Maximum Identifiability Index

By Proposition 6, the maximum identifiability index of a given set S is the minimum per-node maximum identifiability index $\Omega(v)$ for each node $v \in S$. It thus suffices to characterize the per-node maximum identifiability index for each probing mechanism. Under CAP, we give the exact value of $\Omega(v)$ based on the necessary and sufficient condition in Theorem 15; under CSP and UP, we establish tight upper and lower bounds on $\Omega(v)$ based on the conditions in Theorems 18 and 21.

A. Maximum Identifiability Index under CAP

Since Theorem 15 provides necessary and sufficient conditions, it directly determines the value of $\Omega(v)$, as stated below.

Theorem 24 (Maximum Per-node Identifiability under CAP)**.** *The maximum identifiability index of a non-monitor v under CAP is $\Omega^{\text{CAP}}(v) = \Gamma_{\mathcal{G}^*}(v, m')$.*

Evaluation algorithm: As shown in Section IV-A, $\Gamma_{\mathcal{G}^*}(v, m')$ can be computed in $O(\theta\xi)$ time (θ: the number of monitor neighbors in $\mathcal{G}$, ξ: the number of links in $\mathcal{G}$; see Table I). Therefore, $\Omega^{\text{CAP}}(S)$ is computable in $O(\theta\xi|S|)$ time.

B. Maximum Identifiability Index under CSP

Observing that both the sufficient and the necessary conditions in Theorem 18 are imposed on the same property, i.e., vertex-cuts of the auxiliary graph $\mathcal{G}^*$ and $\mathcal{G}_m$. Let $\delta^* := \Gamma_{\mathcal{G}^*}(v, m')$, $\delta_{\min} := \min_{m \in M} \Gamma_{\mathcal{G}_m}(v, m')$, and $\pi_v := \min(\delta_{\min}, \delta^* - 1)$. We obtain a tight characterization of the maximum identifiability index under CSP as follows.

Theorem 25 (Maximum Per-node Identifiability under CSP)**.** *If $\pi_v \leq \sigma - 2$, the maximum identifiability index of a non-monitor v under CSP is bounded by $\pi_v - 1 \leq \Omega^{\text{CSP}}(v) \leq \pi_v$.*

Proof: The proof can be found in [26]. ■

Remark: Because the set of links in $\mathcal{G}_m$ is a subset of those in $\mathcal{G}^*$ while the nodes are the same, we always have $\delta_{\min} \leq \delta^*$. Therefore, the above bounds simplify to:

- $\delta_{\min} - 2 \leq \Omega^{\text{CSP}}(v) \leq \delta_{\min} - 1$ if $\delta_{\min} = \delta^*$;
- $\delta_{\min} - 1 \leq \Omega^{\text{CSP}}(v) \leq \delta_{\min}$ if $\delta_{\min} < \delta^*$.

In particular, if $\delta^* = 1$, then it implies that $\exists$ a node $w \in N$ in $\mathcal{G}^*$, where all simple paths starting at v and terminating at m' must traverse w, i.e., $\nexists$ simple monitor-to-monitor paths traversing v ($P_v = \emptyset$); therefore $\Omega^{\text{CSP}}(v) = 0$ (even single-node failures in S cannot always be localized if $v \in S$).

The only cases when $\pi_v \leq \sigma - 2$ is violated are: (i) $\delta_{\min} = \delta^* = \sigma$, or (ii) $\delta_{\min} = \sigma - 1$ and $\delta^* = \sigma$. In case (i), non-monitor v still has a monitor as a neighbor after removing m; by Proposition 19, this implies that $\Omega^{\text{CSP}}(v) = \sigma$. In case (ii),

Theorem 18 (a) can still be applied to show that $\Omega^{\text{CSP}}(S) \geq \sigma - 2$, and one can verify that the condition in Proposition 19 is violated, which implies that $\Omega^{\text{CSP}}(v) \leq \sigma - 1$. In fact, we can leverage Proposition 20 to uniquely determine $\Omega^{\text{CSP}}(S)$ in this case. If the conditions in Proposition 20 are satisfied, then $\Omega^{\text{CSP}}(v) = \sigma - 1$; otherwise, $\Omega^{\text{CSP}}(v) = \sigma - 2$.

Evaluation algorithm: Evaluating $\Omega^{\text{CSP}}(S)$ by Proposition 6 involves computing $\Omega(v)$ for all $v \in S$, each requiring the computation of the vertex-cuts of the auxiliary graphs $\mathcal{G}^*$ and $\mathcal{G}_m$ ($\forall m \in M$) as that in Section IV-A, which altogether takes $O(\mu\theta\xi|S|)$ time.

C. Maximum Identifiability Index under UP

As in the case of CSP, we can leverage the sufficient and the necessary conditions in Theorem 21 to bound the maximum identifiability index under UP from both sides. The conditions in Theorem 21 imply the following bounds on the maximum identifiability index under UP.

Theorem 26 (Maximum Per-node Identifiability under UP)**.** *The maximum identifiability index of a non-monitor v under UP with measurement paths P is bounded by MSC$(v) - 1 \leq \Omega^{\text{UP}}(v) \leq$ MSC(v).*

Proof: The proof can be found in [26]. ■

Evaluation algorithm: The original bounds in Theorem 26 are hard to evaluate due to the NP-hardness of computing MSC$(\cdot)$. As in Section IV-C, we resort to the greedy algorithm, which implies the following relaxed bounds:

$$\left\lceil \frac{\text{GSC}(v)}{\log(|P_v|) + 1} \right\rceil - 1 \leq \Omega^{\text{UP}}(v) \leq \text{GSC}(v). \tag{5}$$

Evaluating these bounds for $\Omega^{\text{UP}}(S)$ involves invoking the greedy algorithm for each node in S, with an overall complexity of $O(|S||P|^2\sigma)$ (or $O(\mu^4\sigma|S|)$ if all monitors can probe each other).

VI. Characterization of the Maximum Identifiable Set

By Proposition 8, the maximum k-identifiable set $S^*(k)$ is related to the per-node maximum identifiability index $\Omega(v)$ by $S^*(k) = \{v \in N : \Omega(v) \geq k\}$. Therefore, $S^*(k)$ can be easily computed based on values of $\Omega(v)$ ($v \in N$) for any value of k. Moreover, given upper/lower bounds on $\Omega(v)$, i.e., $\Omega_l(v) \leq \Omega(v) \leq \Omega_u(v)$, $S^*(k)$ can be bounded by $S^{\text{inner}}(k) \subseteq S^*(k) \subseteq S^{\text{outer}}(k)$ for $S^{\text{inner}}(k) := \{v \in N : \Omega_l(v) \geq k\}$ and $S^{\text{outer}}(k) := \{v \in N : \Omega_u(v) \geq k\}$. Based on this observation, we now characterize $S^*(k)$ for each of the three probing mechanisms.

A. Maximum k-identifiable Set under CAP

The expression of the maximum per-node identifiability under CAP in Theorem 24 leads to the following characterization of the maximum k-identifiable set.

Corollary 27. *The maximum k-identifiable set under CAP, denoted by $S^*_{CAP}(k)$, is $S^*_{CAP}(k) = \{v \in N : \Gamma_{\mathcal{G}^*}(v, m') \geq k\}$.*

Specifically, when $k = \sigma$, $S^*_{\text{CAP}}(\sigma)$ contains all the non-monitors directly adjacent to monitors.

Evaluation algorithm: As shown in Section IV-A, $\Gamma_{\mathcal{G}^*}(v, m')$ can be computed in $O(\theta\xi)$ time. Thus, the total time complexity for constructing $S^*_{\text{CAP}}(k)$ is $O(\theta\xi\sigma)$.

B. Maximum k-identifiable Set under CSP

Leveraging Theorem 25, we can establish outer and inner bounds (i.e., superset and subset) for the maximum k-identifiable set under CSP.

Corollary 28. *Let $S^{outer}_{CSP}(k) := \{v \in N : \pi_v \geq k\}$, and $S^{inner}_{CSP}(k) := \{v \in N : \pi_v \geq k+1\}$. The maximum k-identifiable set under CSP ($k \leq \sigma - 1$), denoted by $S^*_{CSP}(k)$, is bounded by $S^{inner}_{CSP}(k) \subseteq S^*_{CSP}(k) \subseteq S^{outer}_{CSP}(k)$.*

Proof: The proof can be found in [26]. ■

One case not covered by Corollary 28 is $k = \sigma$. In this case, $S^*_{\text{CSP}}(\sigma)$ contains all non-monitors that have at least two monitors as neighbors according to Proposition 19. Another non-covered case is $k = \sigma - 1$, for which we have the following result.

Corollary 29. *When $k = \sigma - 1$, $S^*_{CSP}(k) = \{v \in N : v$ has at least two monitor neighbors$\} \cup \widetilde{S}$. Set $\widetilde{S}$ contains one and only one non-monitor w if all nodes in N but w have at least two monitor neighbors and w has one monitor and all nodes in $N \setminus \{w\}$ as neighbors; otherwise, $\widetilde{S} = \emptyset$.*

Proof: The proof can be found in [26]. ■

Corollary 29 implies that when $\widetilde{S}$ is not empty (i.e., $|\widetilde{S}| = 1$), then $S^*_{\text{CSP}}(\sigma - 1) = N$ and $S^*_{\text{CSP}}(\sigma) = N \setminus \widetilde{S}$ (i.e., $|S^*_{\text{CSP}}(\sigma - 1)| = \sigma$ and $|S^*_{\text{CSP}}(\sigma)| = \sigma - 1$).

Evaluation algorithm: Corollary 29 is computable in linear time. Similar to Section IV-B, π_v in Corollary 28 is in $O(\mu\theta\xi)$ complexity. Therefore, the overall complexity is $O(\mu\theta\xi\sigma)$.

C. Maximum k-identifiable Set under UP

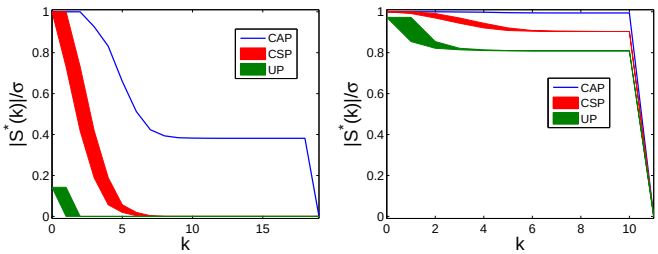
Analogous to the case of CSP, we leverage Theorem 26 to develop the following outer and inner bounds for the maximum k-identifiable set under UP.

Corollary 30. *Let $S^{outer}_{UP}(k) := \{v \in N :$ MSC$(v) \geq k\}$ and $S^{inner}_{UP}(k) := \{v \in N :$ MSC$(v) \geq k + 1\}$ with measurement paths P. The maximum k-identifiable set under UP ($k \leq \sigma - 1$), denoted by $S^*_{UP}(k)$, is bounded by $S^{inner}_{UP}(k) \subseteq S^*_{UP}(k) \subseteq S^{outer}_{UP}(k)$.*

Proof: The proof can be found in [26]. ■

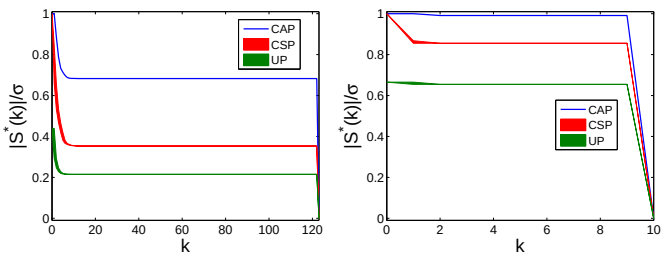
A special case left out by Corollary 30 is $k = \sigma$. In this case, we use Proposition 22 to determine $S^*_{\text{UP}}(\sigma)$, i.e., $S^*_{\text{UP}}(\sigma) = \{w \in N : w$ is on a 2-hop path$\}$.

Evaluation algorithm: Due to the NP-hardness of computing MSC$(\cdot)$, we again resort to the greedy algorithm, whereby the outer and inner bounds of $S^*_{\text{UP}}(k)$ can be relaxed by computing GSC$(\cdot)$. Let $\widehat{S^{\text{outer}}_{\text{UP}}}(k) := \{v \in N : \text{GSC}(v) \geq k\}$ and $\widehat{S^{\text{inner}}_{\text{UP}}}(k) := \{v \in N : \text{GSC}(v)/\big(\log(|P_v|) + 1\big) \geq k + 1\}$. We have $S^{\text{outer}}_{\text{UP}}(k) \subseteq \widehat{S^{\text{outer}}_{\text{UP}}}(k)$ and $\widehat{S^{\text{inner}}_{\text{UP}}}(k) \subseteq S^{\text{inner}}_{\text{UP}}(k)$ according to Proposition 8. The computation of these relaxed bounds involves $O(\sigma|P|^2)$ time complexity w.r.t. each node in N. Thus, the overall complexity is $O(\sigma^2|P|^2)$.



(a) $\mu = 2$ (b) $\mu = 10$

Fig. 4. Maximum k-identifiable set $S^*(k)$ under CAP, CSP, and UP for ER graphs ($|V| = 20$, $\mu = \{2, 10\}$, $\mathbb{E}[|L|] = 51$, 200 graph instances, σ: total number of non-monitors).



(a) $\mu = 50$ (b) $\mu = 163$

Fig. 5. Maximum k-identifiable set $S^*(k)$ under CAP, CSP, and UP for Rocketfuel AS1755 ($|V| = 172$, $|L| = 381$, $\mu = \{50, 163\}$, 100 Monte Carlo runs, σ: total number of non-monitors).

VII. Evaluation of Failure Localization Capability

We demonstrate how the proposed measures of maximum identifiability index and maximum identifiable set can be used to evaluate the impact of various parameters, including topology, number of monitors, and probing mechanisms (CAP, CSP, UP), on the capability of failure localization. In this study, we assume (hop count-based) shortest path routing as the default routing protocol under UP, i.e., measurement paths under UP are the shortest paths between monitors, with ties broken arbitrarily.

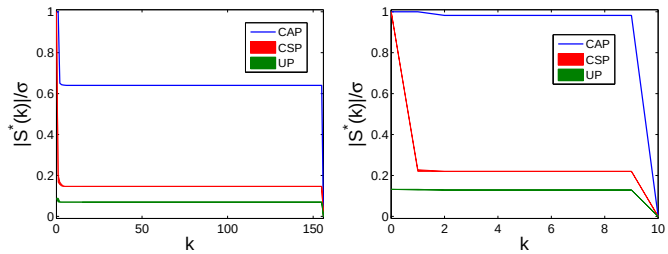
A. Topologies for Evaluation

We first employ random graph models to generate a comprehensive set of topologies without artifacts of specific network deployments. We consider random Erdös-Rényi (ER) graphs [31], generated by independently connecting each pair of nodes by a link with a fixed probability p. The result is a purely random topology where all graphs with an equal number of links are equally likely to be selected (note that the number of nodes is an input parameter). In addition to ER graphs, other random graph models are also considered; the corresponding results are presented in [26] due to space limitation.

We then evaluate real *Autonomous System* (AS) topologies collected by the Rocketfuel [32] and the CAIDA [33] projects, which represents IP-level connections between backbone/gateway routers of several ASes from major *Internet Service Providers (ISPs)* around the globe.

B. Evaluation Results

We focus on evaluating per-node maximum identifiability index $\Omega(v)$ since it determines both the per-set maximum



(a) $\mu = 200$ (b) $\mu = 346$

Fig. 6. Maximum k-identifiable set $S^*(k)$ under CAP, CSP, and UP for CAIDA AS26788 ($|V| = 355$, $|L| = 483$, $\mu = \{200, 346\}$, 100 Monte Carlo runs, σ: total number of non-monitors).

identifiability index $\Omega(S)$ and the maximum identifiable set $S^*(k)$. In particular, the *complementary cumulative distribution function (CCDF)* of $\Omega(v)$ over all $v \in N$ (refer to Table I for notations) coincides with the normalized cardinality of the maximum identifiable set $|S^*(k)|/\sigma$, and thus we characterize the distribution of $\Omega(v)$ by evaluating $|S^*(k)|/\sigma$ wrt k. Moreover, we examine the specific value of $\Omega(v)$ and compare it with the degree (i.e., number of neighbors) of v among monitor/non-monitor nodes to evaluate the correlation between maximum identifiability index and the graph property (i.e., degree) of a node. Under UP, our extensive simulations under multiple graph models have shown that $\text{MSC}(v)$ can be closely approximated by $\text{GSC}(v)$; hence, we use $\text{GSC}(v)$ in place of $\text{MSC}(v)$ for computing Ω^{UP} and S^*_{UP}; see [26] for details.

1) Distribution of $\Omega(v)$: To characterize the overall distribution of $\Omega(v)$, we compute (bounds on)[8] $S^*_{\text{CAP}}(k)$, $S^*_{\text{CSP}}(k)$, and $S^*_{\text{UP}}(k)$ to evaluate $|S^*(k)|/\sigma$ for different values of k (σ: total number of non-monitors). Fig. 4 reports averages of $|S^*(k)|/\sigma$ computed on ER graphs over randomly selected multiple instances of topology and monitor locations, where $|S^*(k)|/\sigma$ under CSP and UP is represented by a band with its width determined by $(|S^{\text{outer}}(k)| - |S^{\text{inner}}(k)|)/\sigma$. The results show large differences in the failure localization capabilities of different probing mechanisms: When the number of monitors is small ($\mu = 2$) and $k = 2$, $S^*_{\text{UP}}(k)$ is almost empty, i.e., no (non-monitor) node state can be uniquely determined by UP when there are multiple failures; in contrast, $|S^*_{\text{CSP}}(k)|/\sigma \approx 0.5$ and $|S^*_{\text{CAP}}(k)|/\sigma \approx 1$, i.e., CSP can uniquely determine the states of half of the nodes and CAP can determine the states of all the nodes when $\mu = 2$ and $k = 2$. When the number of monitors increases ($\mu = 10$), there exist more measurement paths between monitors, and thus the fraction of identifiable nodes increases for all three probing mechanisms. In addition, we observe a stable phase in Fig. 4 where the value of $|S^*(k)|/\sigma$ remains the same as we increase k; this is because some non-monitors have monitors as neighbors, thus directly measurable by these neighboring monitors without traversing other non-monitors. Specifically, if there are non-monitors that neighbor at least one monitor under CAP, neighbor at least two monitors under CSP, or lie on 2-hop paths between monitors under UP, then the failure of these non-monitors can always be identified regardless of the total number of failures in the network, i.e., the maximum identifiability index of these non-monitors is the total number of non-monitors. Note that in

[8] Propositions 19, Corollary 29, and Proposition 22 are used to determine the exact elements in $S^*_{\text{CSP}}(\sigma)$, $S^*_{\text{CSP}}(\sigma - 1)$, and $S^*_{\text{UP}}(\sigma)$.

Fig. 4, the number of such directly measurable non-monitors is smaller under UP than under CSP. This is because for non-monitors that neighbor the same pair of monitors (e.g., m_1 and m_2), all these non-monitors are directly measurable on 2-hop m_1-to-m_2 paths under CSP; however, only one of these non-monitors is on a 2-hop m_1-to-m_2 path under UP as UP probes only one routing path between each pair of monitors (assuming stable single-path routing). Similar results have been obtained for other random graph models (see [26] for details).

We repeat the above evaluation on AS topologies. We select AS1755 from Rocketfuel topologies [32] and AS26788 from CAIDA topologies [33], and evaluate the bounds on $|S^*(k)|/\sigma$ under multiple instances of random monitor placements; average results are reported in Fig. 5 and 6. Similar to the case of random topologies, there are clear differences between different probing mechanisms. Unlike the uniformly connected random topologies in Fig. 4, these AS topologies contain many sparse subgraphs where the removal of a few nodes can disconnect the network. Thus, unless a node is directly measurable by monitors, it is likely that failures of a few other nodes will disconnect it from monitors and thus make its failure undetectable. Comparing results from Rocketfuel and CAIDA, we observe that the CAIDA AS requires more monitors to achieve the same level of identifiability. Moreover, deploying more monitors in CAIDA AS only slightly improves the level of identifiability. This can be explained by examining the link density $|L|/|V|$ of the network: $|L|/|V| = 1.36$ for the CAIDA AS, whereas $|L|/|V| = 2.22$ for the Rocketfuel AS, i.e., CAIDA AS topology is nearly a tree. Therefore, it is likely for a node to not reside on any paths between monitors or become unmeasurable after the failure of one other node in the CAIDA AS, even if the paths are controllable but cycle-free (CSP).

2) Correlation of $\Omega(v)$ and Degree: Next, we examine specific values of $\Omega(v)$ for each non-monitor $v \in N$ for selected instances of network topology and monitor placement. Our goal is to compare these values with node degrees to understand the correlation between the proposed identifiability measure and typical graph-theoretic node properties. Specifically, we sort non-monitors in a non-increasing order of $\Omega(v)$ under each of the three probing mechanisms, and compare $\Omega(v)$ with the degrees of v among monitors/non-monitors[9]; see results in Fig. 7 for random topologies and in Fig. 8–9 for AS topologies. The results show strong correlations between $\Omega(v)$ and the degree of v, denoted by $\mathrm{d}(v)$. Specifically, denote the number of neighbors of v that are monitors by $\mathrm{d}^m(v)$ and the number of neighbors of v that are non-monitors by $\mathrm{d}^n(v)$; the overall degree $\mathrm{d}(v) = \mathrm{d}^m(v) + \mathrm{d}^n(v)$. If node v has sufficient monitor neighbors ($\mathrm{d}^m(v) \geq 1$ for CAP, $\mathrm{d}^m(v) \geq 2$ for CSP), then v is directly measurable and thus $\Omega(v) = \sigma$ regardless of the actual degree of v; if node v does not have a sufficient number of monitors as neighbors, then $\Omega(v) \leq \mathrm{d}(v)$ because if all neighbors of v fail, then the state of v cannot be determined by path measurements. However, in the latter case, $\mathrm{d}(v)$ is only a loose upper bound, and the exact value of $\Omega(v)$ depends on the overall topology, the locations of monitors, and the constraints on measurement paths. In this regard, our result can also be viewed as defining a new node property ($\Omega(v)$) that takes into account all these parameters.

[9]Note that node IDs are different under different probing mechanisms due to the different order of $\Omega(v)$ values.

Overall, we observe that CAP-type probing is hugely advantageous in uniquely monitoring node states under failures, especially when there are multiple failures and the network is sparse. This implies that in the absence of deploying monitors at every node, implementing controllable probing is an effective way to uniquely localize node failures. Our observation also stresses the importance of optimized monitor placement, especially when we are only interested in monitoring a subset of nodes, which is left to future work.

VIII. Conclusion

We studied the fundamental capability of a network in localizing failed nodes from binary measurements (normal/failed) of paths between monitors. We proposed two novel measures: *maximum identifiability index* that quantifies the scale of uniquely localizable failures wrt a given node set, and *maximum identifiable set* that quantifies the scope of unique localization under a given scale of failures. We showed that both measures are functions of the maximum identifiability index per node. We studied these measures for three types of probing mechanisms that offer different controllability of probes and complexity of implementation. For each probing mechanism, we established necessary/sufficient conditions for unique failure localization based on network topology, placement of monitors, constraints on measurement paths, and scale of failures. We further showed that these conditions lead to tight upper/lower bounds on the maximum identifiability index, as well as inner/outer bounds on the maximum identifiable set. We showed that both the conditions and the bounds can be evaluated efficiently using polynomial-time algorithms. Our evaluations on random and real network topologies showed that probing mechanisms that allow monitors to control the routing of probes have significantly better capability to uniquely localize failures.

References

[1] R. Kompella, J. Yates, A. G. Greenberg, and A. C. Snoeren, "Detection and localization of network black holes," in *IEEE INFOCOM*, 2007.

[2] M. Coates, A. O. Hero, R. Nowak, and B. Yu, "Internet tomography," *IEEE Signal Processing Magazine*, vol. 19, pp. 47–65, 2002.

[3] D. Ghita, C. Karakus, K. Argyraki, and P. Thiran, "Shifting network tomography toward a practical goal," in *ACM CoNEXT*, 2011.

[4] Y. Bejerano and R. Rastogi, "Robust monitoring of link delays and faults in IP networks," in *IEEE INFOCOM*, 2003.

[5] J. D. Horton and A. López-Ortiz, "On the number of distributed measurement points for network tomography," in *ACM IMC*, 2003.

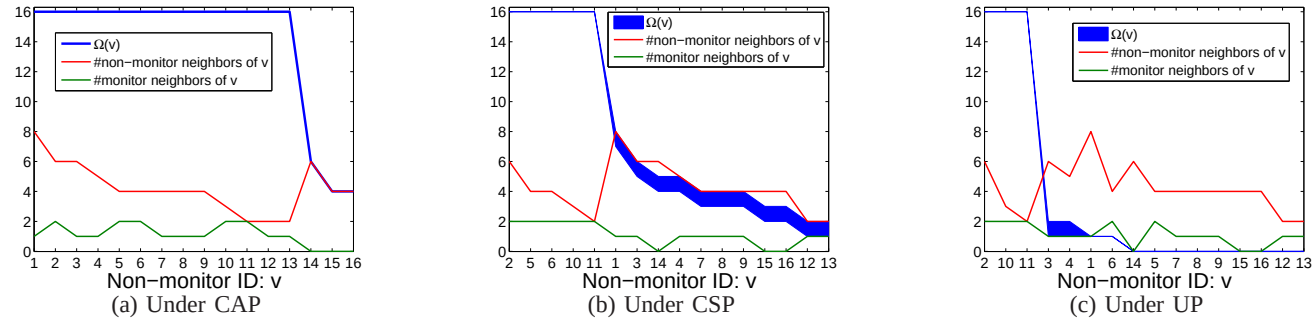
[6] S. Zarifzadeh, M. Gowdagere, and C. Dovrolis, "Range tomography: Combining the practicality of boolean tomography with the resolution of analog tomography," in *ACM IMC*, 2012.

[7] A. Markopoulou, G. Iannaccone, S. Bhattacharyya, C.-N. Chuah, and C. Diot, "Characterization of failures in an IP backbone," in *IEEE INFOCOM*, 2004.

[8] N. Duffield, "Simple network performance tomography," in *ACM IMC*, 2003.

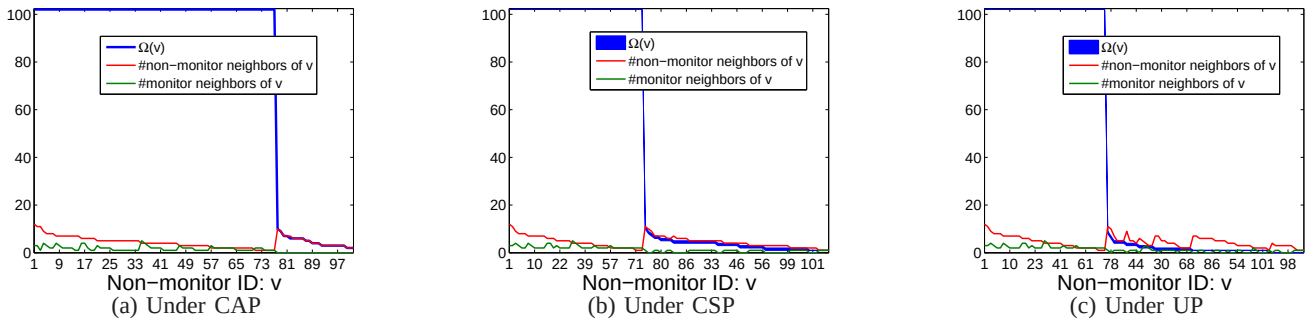
[9] ——, "Network tomography of binary network performance characteristics," *IEEE Transactions on Information Theory*, vol. 52, pp. 5373–5388, 2006.

[10] H. Zeng, P. Kazemian, G. Varghese, and N. McKeown, "Automatic test packet generation," in *ACM CoNEXT*, 2012.



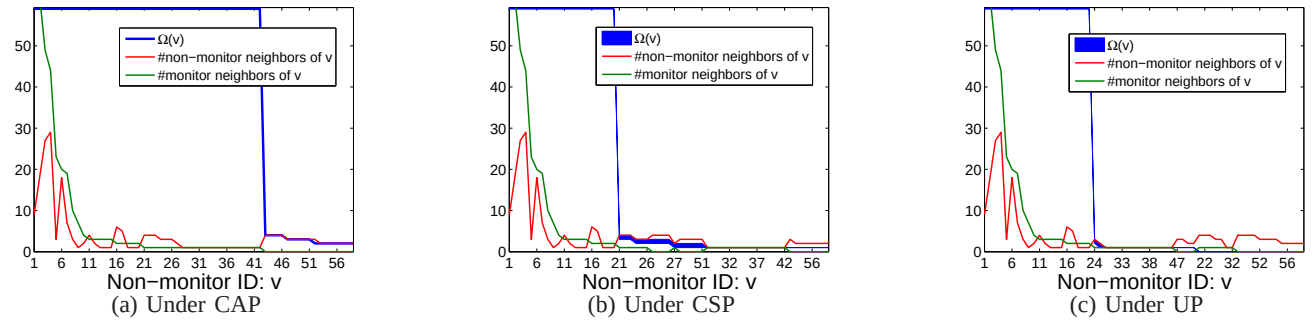
(a) Under CAP (b) Under CSP (c) Under UP

Fig. 7. Node maximum identifiability index $\Omega(v)$ of one ER graph under different probing mechanisms ($|V| = 20$, $\mu = 4$, $\mathbb{E}[|L|] = 51$).



(a) Under CAP (b) Under CSP (c) Under UP

Fig. 8. Node maximum identifiability index $\Omega(v)$ of Rocketfuel AS1755 under different probing mechanisms ($|V| = 172$, $|L| = 381$, $\mu = 70$).



(a) Under CAP (b) Under CSP (c) Under UP

Fig. 9. Node maximum identifiability index $\Omega(v)$ of CAIDA under different probing mechanisms ($|V| = 355$, $|L| = 483$, $\mu = 296$).

[11] H. Nguyen and P. Thiran, "The boolean solution to the congested IP link location problem: Theory and practice," in *IEEE INFOCOM*, 2007.

[12] A. Dhamdhere, R. Teixeira, C. Dovrolis, and C. Diot, "Netdiagnoser: Troubleshooting network unreachabilities using end-to-end probes and routing data," in *ACM CoNEXT*, 2007.

[13] Y. Huang, N. Feamster, and R. Teixeira, "Practical issues with using network tomography for fault diagnosis," *ACM SIGCOMM Computer Communication Review*, vol. 38, pp. 53–58, 2008.

[14] H. X. Nguyen and P. Thiran, "Active measurement for multiple link failures diagnosis in IP networks," in *PAM*, 2004.

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

[16] S. Cho and S. Ramasubramanian, "Localizing link failures in all-optical networks using monitoring tours," *Elsevier Computer Networks*, vol. 58, pp. 2–12, 2014.

[17] L. Ma, T. He, A. Swami, D. Towsley, K. K. Leung, and J. Lowe, "Node failure localization via network tomography," in *ACM IMC*, 2014.

[18] M. Cheraghchi, A. Karbasi, S. Mohajer, and V. Saligrama, "Graph-constrained group testing," *IEEE Transactions on Information Theory*, vol. 58, pp. 248–262, 2012.

[19] [Online]. Available: http://www.ietf.org/rfc/rfc0791.txt

[20] "Open networking foundation." [Online]. Available: http://www.opennetworkingfoundation.org

[21] "OpenFlow switch specification," Open Networking Foundation, Version 1.4.0, October 2013. [Online]. Available: https://www.opennetworking.org/images/stories/downloads/sdn-resources/onf-specifications/openflow/openflow-spec-v1.4.0.pdf

[22] [Online]. Available: https://www.ietf.org/rfc/rfc3031.txt

[23] S. Ahuja, S. Ramasubramanian, and M. Krunz, "Single-link failure detection in all-optical networks using monitoring cycles and paths," *IEEE/ACM Transactions on Networking*, vol. 17, pp. 1080–1093, 2009.

[24] R. Dorfman, "The detection of defective members of large populations," *The Annals of Mathematical Statistics*, vol. 14, 1943.

[25] H.-G. Yeh, "d-Disjunct matrices: Bounds and Lovasz local lemma," *Discrete Math*, vol. 253, pp. 97–107, 2002.

[26] L. Ma, T. He, A. Swami, D. Towsley, and K. K. Leung, "Failure localization capability: Theorem proof and evaluation," Technical Report, IBM T.J. Watson Research, New York, March 2015. [Online]. Available: http://resweb.watson.ibm.com/researcher/files/us-maliang/TechnicalReportTON2015.pdf

[27] J. Chuzhoy and S. Khanna, "Polynomial flow-cut gaps and hardness of directed cut problems," *Journal of the ACM*, vol. 56, pp. 1–28, 2009.

[28] L. R. Ford and D. R. Fulkerson, "Maximal flow through a network," *Canadian Journal of Mathematics*, vol. 8, pp. 399–404, 1956.

[29] V. Chvatal, "A greedy heuristic for the set-covering problem," *Mathematics of Operations Research*, vol. 4, pp. 233–235, 1979.

[30] R. Tarjan, "Depth-first search and linear graph algorithms," *SIAM Journal on Computing*, vol. 1, pp. 146–160, 1972.

[31] P. Erdös and A. Rényi, "On the evolution of random graphs," *Publications of the Mathematical Institute of the Hungarian Academy of Sciences*, vol. 5, pp. 17–61, 1960.

[32] "Rocketfuel: An ISP topology mapping engine," University of Washington, 2002. [Online]. Available: http://www.cs.washington.edu/research/networking/rocketfuel/

[33] "Macroscopic Internet Topology Data Kit (ITDK)," The Cooperative Association for Internet Data Analysis (CAIDA), April 2013. [Online]. Available: http://www.caida.org/data/active/internet-topology-data-kit/