Node Failure Localization in Communication Networks via Network Tomography

Liang Ma\textsuperscript{\S}, Ting He\textsuperscript{\P}, Ananthram Swami\textsuperscript{\D}, Don Towsley\textsuperscript{\*,} Kin K. Leung\textsuperscript{\†}, and Jessica Lowe\textsuperscript{\‡}

\textsuperscript{\S}Imperial College, London, UK. Email: \{lma10, kin.leung\}@imperial.ac.uk
\textsuperscript{\D}IBM T. J. Watson Research Center, Yorktown, NY, USA. Email: the@us.ibm.com
\textsuperscript{\P}Army Research Laboratory, Adelphi, MD, USA. Email: ananthram.swami.civ@mail.mil
\textsuperscript{\*}University of Massachusetts, Amherst, MA, USA. Email: towsley@cs.umass.edu
\textsuperscript{\‡}DSTL, Salisbury, UK. Email: jjlowe@dstl.gov.uk

Abstract—We investigate the problem of localizing node failures in a communication network from end-to-end path measurements, under the assumption that a path behaves normally if and only if it does not contain any failed nodes. To uniquely localize node failures, the measurement paths must show different symptoms under different failure events, i.e., for any two distinct sets of failed nodes, there must be a measurement path traversing one and only one of them. This condition is, however, impractical to test for large networks due to the combinatorial numbers of paths and failure sets. Our first contribution is a characterization of this condition in terms of easily verifiable conditions on the network topology with given monitor placements under three families of probing mechanisms, which differ in whether measurement paths are (i) arbitrarily controllable, (ii) controllable but cycle-free, or (iii) uncontrollable (i.e., determined by the default routing protocol). Our second contribution is a characterization of the maximum identifiability of node failures, measured by the maximum number of simultaneous failures that can always be uniquely localized. Specifically, we bound the maximal identifiability from both the upper and the lower bounds which differ by at most one, and show that these bounds can be evaluated in polynomial time. Finally, we quantify the impact of the probing mechanism on the capability of node failure localization by numerically comparing the maximum identifiability under different probing mechanisms on both random and real network topologies. We observe that despite a higher implementation cost, probing along controllable paths can significantly improve a network’s capability to localize simultaneous node failures.

I. INTRODUCTION

Effective monitoring of network performance is essential for coalition networks in building a reliable network substrate that is robust against service disruptions. In order to achieve this goal, the monitoring infrastructure must be able to detect network misbehaviors (e.g., large delay, unreachable) and localize the sources of the anomaly in an accurate and timely manner. Knowledge of where problematic network elements reside in the network is particularly useful for fast service recovery. For example, if the service disruption is caused by unavailability/overloading at the server, then the network administrator needs to migrate the service to backup servers, but if the disruption is caused by malfunction of certain forwarding nodes, then the network administrator needs to reroute traffic to avoid the affected nodes. Localizing network elements that cause a service disruption can be challenging in hybrid coalition networks. In such networks, a successful end-to-end service for one coalition partner often involves elements in the domains of other partners and/or third parties (e.g., civilian cellular/cloud networks). However, it is not always feasible to directly monitoring the health of these elements due to system heterogeneity and/or constraints imposed by the domain administrators for security/bandwidth reasons. These limitations call for a different approach that can diagnose the health of network elements from the health of end-to-end connections perceived between a subset of elements capable of taking measurements (e.g., gateway nodes).

This different approach is generally known as network tomography [2]. Network tomography provides a methodology for inferring internal network characteristics by measuring end-to-end performance from a subset of nodes with monitoring capabilities, referred to as monitors. Unlike direct measurements, network tomography only relies on end-to-end performance (e.g., path connectivity) experienced by data packets, thus addressing issues such as overhead and lack of internal cooperation in hybrid networks. In cases where the network characteristic of interest is binary (e.g., normal or failed), the problem 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. Assuming 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 cannot determine which one. Observing that there are often multiple explanations for given path failures, existing work mostly focuses on finding the most probable explanation that involves the minimum set of failed nodes. There is, however, no 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 settings such as topology, monitor placement, and measurement routing.

In this paper, we consider two closely related problems: In a network with given monitor placement, (1) if the number...
of simultaneous node failures is bounded by \( k \), then under what conditions can one uniquely localize failed nodes from path measurements? (2) what is the maximum number of simultaneous node failures (i.e., the largest value of \( k \)) that can be uniquely localized in this network? We study both problems in the context of the following families of probing mechanisms: (i) **Controllable Arbitrary-path Probing (CAP)**, where measurement paths are arbitrarily controllable, (ii) **Controllable Simple-path Probing (CSP)**, where measurement paths are controllable but 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 the routing of probing packets and are feasible in different network scenarios (see Section II-C); answers to the above two 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.

In the sequel, we assume that node failures are persistent, i.e., a failed node remains failed throughout the measurement process and leads to failures of all paths traversing it.

### A. Related Work

Little is known for uniquely localizing network failures. Given a set of monitors known to uniquely localize failures on paths between themselves, [4] 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, [5] 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. However, the condition remains unknown if the failures are associated with nodes instead of links, or constraints (e.g., cycle-free) are imposed on measurement paths by the routing protocols. In this paper, we investigate the fundamental relationships between node failure identifiability and explicit network settings such as topology, placement of monitors, and probing mechanism, with focus on developing efficient algorithms to characterize the capability of failure localization under given settings.

### B. Summary of Contributions

We study, for the first time, the fundamental capability of a network with given monitor placements to uniquely localize node failures from binary end-to-end measurements between monitors. Our contributions are five-fold:

1. We propose a novel measure, referred to as maximum identifiability, to characterize a network’s capability in failure localization as the maximum number of simultaneous node failures it can uniquely localize.

2. We establish concrete conditions in terms of network topology, placement of monitors, and measurement paths under 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 and lower bounds on the maximum identifiability that narrows its value to at most two consecutive integers. The bounds are polynomial-time computable under CAP and CSP; while they are NP-hard to compute under UP, we give a greedy heuristic to compute a pair of relaxed bounds that frequently coincide with the original bounds in practice.

4. We extensively compare the maximum identifiability under different probing mechanisms on random and real topologies. Our comparison shows that although controllable probing, especially CAP, is more difficult to implement, it significantly improves the capability of failure localization in terms of maximum identifiability.

Note that 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 maximum identifiability. We leave the characterization of maximum identifiability in the presence of such additional information to future work.

The rest of the paper is organized as follows. Section II formulates the problem. Section III presents verifiable conditions for specific families of probing mechanisms. Based on the derived conditions, Section IV presents bounds on the maximum identifiability that can be efficiently evaluated. The bounds are evaluated on various synthetic/real topologies in Section V. Finally, Section VI concludes the paper.

### II. PROBLEM FORMULATION

#### A. Models and Assumptions

We assume that the network topology is known and can be modeled as an undirected graph \( \mathcal{G} = (V, L) \), where \( V \) and \( L \) are the sets of nodes and links. In \( \mathcal{G} \), the number of neighbors of node \( v \) is called the degree of \( v \). Without loss of generality, we assume \( \mathcal{G} \) is connected, as different connected components have to be monitored separately.

A subset of nodes \( M \subseteq V \) 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 filtered out within the monitoring system. Non-monitors, on the other hand, may fail, and a failure event may involve simultaneous failures of multiple non-monitors. A failure set \( F \) is a set of non-monitors \( (F \subseteq N) \) that may fail simultaneously. Depending on the adopted probing mechanism, monitors can measure and determine the states of nodes by sending probes along certain paths. Let \( P \) denote the set of all possible measurement paths under a given probing mechanism; for given \( \mathcal{G} \) and \( M \), different probing mechanisms can lead to different sets

---

1We use the terms network and graph interchangeably.
of measurement paths, which will be specified later. We use
node state (path state) to refer to the state, failed or normal, of
nodes (paths), where a path fails if and only if at least one node
on the path fails. To avoid trivial cases, we assume that each
non-monitor is traversed by at least one measurement path, as
otherwise the non-monitor is unobservable to the monitoring
system and thus can be excluded in failure localization. Table I
summarizes graph-related notations used in this paper (follow-
ning the convention of [6]).
Let \( w = (W_1, \ldots, W_{\sigma})^T \) be the binary column vector
of the states of all non-monitors and \( c = (C_1, \ldots, C_{\gamma})^T \)
the binary column vector of the states of all measurement paths.
For both node and path states, 0 represents “normal” and 1
represents “failed”. We can relate the path states to the node
states through the following Boolean linear system:

\[
R \odot w = c,
\]

where \( R = (R_{ij}) \) 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 \( P_i \) (1: yes; 0: no), and “\( \odot \)” is the Boolean matrix
product, i.e., \( C_i = \lor_{j=1}^{\gamma} (R_{ij} \land W_j) \). The goal of Boolean
network tomography is to invert this Boolean linear system
to solve for \( w \) given \( R \) and \( c \). Intuitively, node failures are
identifiable if and only if (1) has a unique solution.

B. Definitions
The challenge in failure localization is that the solution to
(1) is usually not unique, i.e., there are multiple possible failure
sets leading to the observed path states. To reduce ambiguity,
we limit the solution space to a predetermined collection of
likely failure sets and only seek to ensure uniqueness within
this collection. Let \( P_F \) denote the set of all measurement paths
affected by a failure set \( F \) (i.e., traversing at least one node
in \( F \)). We now formally define the notion of identifiability in
node failure localization.

Definition 1. Given a network \( \mathcal{G} \) and a set of measurement
paths \( P \) in \( \mathcal{G} \):

1) Two failure sets \( F_1 \) and \( F_2 \) can be distinguished from
each other 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 \( F_2 \).
2) A collection of failure sets \( \Psi \) is identifiable if and only if
every two different failure sets in \( \Psi \) can be distinguished
from each other.

Intuitively, the identifiability of a set \( F \) in a collection
of sets \( \Psi \) means that we can uniquely localize node failures from
observed path states if \( \Psi \) contains all possible failure sets and
the actual set of failed nodes equals \( \hat{F} \), and the identifiability of
\( \Psi \) means that we can always uniquely localize node failures
if the set of failed nodes falls into \( \Psi \).

It is clear from the definition that whether a failure set is
identifiable or not depends on the collection of potential failure
sets it is compared against. In practice, simultaneous failure of
many nodes is an unlikely event. A natural definition of \( \Psi \) is
therefore to restrict the cardinality of failure sets, which leads
to the following notions.

Definition 2. Given a network \( \mathcal{G} \) and a set of measurement
paths \( P \) in \( \mathcal{G} \):

1) We say \( \mathcal{G} \) is \( k \)-identifiable \((0 \leq k \leq \gamma)\) if the collection
\( \Psi \) of all subsets of \( N \) with cardinality bounded by \( k \)
is identifiable, i.e., any failure of up to \( k \) nodes can be
uniquely localized.
2) The maximum identifiability of \( \mathcal{G} \), denoted by \( \Omega(\mathcal{G}) \),
is the maximum value of \( k \) such that \( \mathcal{G} \) is \( k \)-identifiable.

The maximum identifiability of a network characterizes its
capability to localize failures in the worst case. That is, no
matter where the failures occur, as long as the number of failed
nodes is bounded by \( \Omega \), we can uniquely localize the failures
from observed path states. Note that it is possible to uniquely
localize a larger number of failures when they occur at a
particular set of nodes, but localization cannot be guaranteed
if the failures occur elsewhere. Both \( k \)-identifiability and
maximum identifiability are defined with respect to a given
\( P \), which will be clear from the context.

C. Classification of Probing Mechanisms
Given the topology \( \mathcal{G} \) and the monitor locations \( M \), the
probing mechanism plays a crucial role in failure localization
by determining the set of measurement paths \( P \). Depending
on the flexibility of probing and the cost of deployment, we
partition probing mechanisms into three families:

1) Controllable Arbitrary-path Probing (CAP): \( P \) includes
any path/cycle, allowing repeated nodes/links, as long as
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.

In particular, although CAP allows probes to traverse each
node/link an arbitrary number of times, it suffices for probes
to traverse each link at most once in either direction for the
sake of localizing node failures.

These probing mechanisms clearly provide decreasing flexi-
bility to the monitors and therefore decreasing capability to
localize failures. Further, they also offer decreasing deploy-
ment cost. At the IP layer, CAP is feasible only if (strict)
source routing (an IP option) [7] is enabled at all non-
onmonitors, which allows them to modify the source and the
destination addresses in packet headers hop by hop to probe
a path prescribed by the monitor initiating the measurement
probe\(^2\). If implemented at the application layer (e.g., to localize
failures in overlay networks), CAP requires equivalent “source
routing” to be supported by the application. Similarly, CSP is
feasible under source routing (or equivalent capability at the
application layer). It is also feasible under an emerging net-
working paradigm called software-defined networking (SDN)
[8], where monitors can instruct the SDN controller to set up
arbitrary cycle-free paths for the probing traffic. Note that the
cycle-free constraint is crucial in SDN, as data forwarding is
performed in a distributed manner by switches according to
forwarding tables configured by the controller during route
setup, which will encounter forwarding loops if the path has
cycles. In contrast, UP only requires basic data forwarding and
is generally feasible.

To strike a balance between capability and cost, the key
question is: what is the benefit of a more expensive probing
mechanism such as CAP in localizing failures, over a cheaper
but more restrictive probing mechanism?

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 bound \( k \) on
the number of simultaneous failures, can we uniquely localize up

\(^2\)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.
where traversing one and only one of them. However, these three as for every two non-monitors, there is a measurement path constraints on measurement paths. Under UP, suppose that the examine this capability and how it can be improved by relaxing
monitors’ capability to identify failures of the non-monitors
measurable between the monitors. In this example, we will
monitors’ capability to identify failures of the non-monitors
if CAP is supported, then we can send probes along a cycle
that can be used to determine the state of
and
(yielding an expanded measurement matrix in (3):

\[ P_1 = m_1v_2v_1m_2, \quad P_2 = m_1v_2v_3m_3, \]  
\[ P_3 = m_2v_3m_3, \]  

where \( R_{UP} = I \) if and only if node \( v_j \) is on path \( P_i \). Then we have \( R_{UP} \circ \mathbf{w} = \mathbf{c} \), where \( \mathbf{c} \) is the binary vector of path states observed at the destination monitors. Based on Definition 1, we can verify that any single node failure is identifiable, as for every two non-monitors, there is a measurement path traversing one and only one of them. However, these three paths cannot identify simultaneous failures of two nodes. This is because if node \( v_2 \) fails, then we cannot determine if \( v_1 \) (or \( v_4 \)) fails or not. Identifiability can be improved if more measurement paths are allowed. For example, under CSP, besides the three paths in (2), we can probe three additional paths: \( P_4 = m_1v_2v_3m_2, P_5 = m_1v_1m_2, \) and \( P_6 = m_2v_4m_3, \) yielding an expanded measurement matrix in (3):

\[ P_1 = m_1v_2v_1m_2, \quad P_2 = m_1v_2v_3m_3, \quad P_3 = m_2v_3m_3, \quad P_4 = m_1v_2v_3m_2, \quad P_5 = m_1v_1m_2, \quad P_6 = m_2v_4m_3, \]  

Using the six paths in (3), we can identify up to three failed nodes, a notable improvement over UP. However, if \( v_1, v_3, \) and \( v_4 \) all fail, then there is no measurement path under CSP that can be used to determine the state of \( v_2 \). Nevertheless, if CAP is supported, then we can send probes along a cycle \( P_7 = m_1v_2m_3 \). In conjunction with the paths in (3), this yields the measurement matrix in (4):

\[ P_1 = m_1v_2m_2, \quad P_2 = m_1v_2m_3, \quad P_3 = m_2v_3m_3, \quad P_4 = m_1v_2m_3, \quad P_5 = m_1v_1m_2, \quad P_6 = m_2v_4m_3, \]  

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

III. VERIFIABLE IDENTIFIABILITY CONDITIONS

In this section, we develop the concrete conditions suitable for efficient testing for the three families of probing mechanisms in Section II-C.

A. Conditions under CAP

Under CAP, we can essentially “ping” any node from a monitor along any path. In the face of failures, this implies that a monitor’s ability to determine the state of a node depends on its connectivity to the monitors after removing nodes that are known/hypothesized to have failed.

**Theorem 3** (k-identifiability under CAP). Network \( G \) is k-identifiable under CAP:

a) if for any set \( V' \) of up to \( k \) non-monitors, each connected component in \( G - V' \) contains a monitor;

b) only if for any set \( V' \) of up to \( k - 1 \) non-monitors, each connected component in \( G - V' \) contains a monitor.

**Proof.** See [9].

Simple as they look, these conditions still cannot be tested efficiently because they enumerate over a combinatorial number of sets \( V' \). Fortunately, we are able to reduce them into explicit conditions on the vertex-connectivity of a related topology, which can then be tested in polynomial time. We use the following notions from graph theory.

**Definition 4.** [6] Graph \( G \) of \( |V| \) vertices is said to be k-vertex-connected if \( k \leq |V| - 1 \) and deleting any subset of up to \( k - 1 \) vertices does not disconnect \( G \). The vertex-connectivity of \( G \), denoted by \( \delta(G) \), is the maximum \( k \) such that \( G \) is k-vertex-connected.

In our problem, the key observation is that requiring each connected component in \( G - V' \) to contain a monitor is equivalent to requiring each connected component in \( G - M - V' \) (i.e., after removing all monitors) to contain a neighbor of a monitor. Thus, if we add virtual links between these neighbors, the resulting graph \( G - M - V' + L(N(M), N(M)) \) should be connected. However, this does not mean the conditions are equivalent, because if \( G - M - V' \) is already connected, \( G - M - V' + L(N(M), N(M)) \) will certainly be connected but \( G - M - V' \) may not contain any neighbors of monitors. This special case can be avoided by introducing a virtual monitor \( m' \) connected to all neighbors of monitors via virtual links, resulting in an auxiliary graph \( G^* := G - M + \{m'\} + L(N(M), N(M)) + L(m', N(M)) \) as illustrated in Fig. 2 (b). We will show that requiring at least one monitor per connected component in \( G - V' \) is equivalent to requiring \( G^* - V' \) to be connected.

The beauty of this new condition is that it reduces the tests over all possible \( V' \) to a single test of the vertex-connectivity of \( G^* \), as stated below.

**Lemma 5.** Each connected component in \( G - V' \) contains a monitor for any set \( V' \) of up to \( s \) \((s \leq \sigma - 1)\) non-monitors if and only if \( G^* \) is \((s + 1)\)-vertex-connected.

**Proof.** See [9].

Lemma 5 allows us to rewrite the identifiability conditions in Theorem 3 in terms of the vertex-connectivity of \( G^* \).
Theorem 8 (k-identifiability under CSP). Network $\mathcal{G}$ 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 $\mathcal{G} - V'$ contains a monitor;
b) only if for any node set $V'$, $|V'| \leq k$, containing at most one monitor, each connected component in $\mathcal{G} - V'$ contains a monitor.

Proof. See [9].

Testing algorithm: A key advantage of the derived conditions is that they can be tested efficiently. Given a value of $k$, we can evaluate the vertex-connectivity of $\mathcal{G}^*$, $\delta(\mathcal{G}^*)$, by the algorithm for determining network vertex connectivity in [10] in $O(\sigma^{0.75})$ time and compare the result with $k + 1$ or $k$ to test the conditions in Corollary 6.

B. Conditions under CSP

Under CSP, we restrict measurement paths $P$ to be the set of simple paths between monitors, i.e., paths starting/ending at distinct monitors and containing no cycles. We give the following result analogous to Theorem 3.

Lemma 5. We will show that the last condition is equivalent to requiring $G_m^* - F$ to be connected, and thus the following holds.

Lemma 6. The following two conditions are equivalent:
(1) Each connected component in $\mathcal{G} - V'$ contains a monitor for any set $V'$ consisting of monitor $m \in M$ and up to $s$ $(s \leq \sigma - 1)$ non-monitors;
(2) $G_m^*$ is $(s + 1)$-vertex-connected.

Proof. See [9].

Based on Lemmas 5 and 9, we can rewrite Theorem 8 as follows.

Corollary 10. Network $\mathcal{G}$ is k-identifiable under CSP:
a) if $G^*$ is $(k + 2)$-vertex-connected, and $G_m^*$ is $(k + 1)$-vertex-connected for each monitor $m \in M$ ($k \leq \sigma - 2$);
b) only if $G^*$ is $(k+1)$-vertex-connected, and $G_m^*$ is k-vertex-connected for each monitor $m \in M$ ($k \leq \sigma - 1$).

Special cases left out by this corollary are the cases of $k = \sigma$ and $k = \sigma - 1$, which are addressed separately as follows.

Corollary 11. Network $\mathcal{G}$ is $\sigma$-identifiable under CSP if and only if each non-monitor has at least two monitors as neighbors.

Proof. See [9].

Corollary 12. Network $\mathcal{G}$ is $(\sigma - 1)$-identifiable under CSP if and only if all but one non-monitor, denoted by $v$, have at least two monitors as neighbors, and $v$ either has (i) two or more monitors as neighbors, or (ii) one monitor and all the other non-monitors (i.e., $N \setminus \{v\}$) as neighbors.

Proof. See [9].

Testing algorithm: Similar to the case of CAP, we can use the algorithm in [10] to compute the vertex-connectivities of the auxiliary graphs $G^*$ and $G_m^*$ ($\forall m \in M$), and then compare the results with $k + 2$ and $k + 1$ (or $k + 1$ and $k$) to test the conditions in Corollary 10 for any given $k$. The overall complexity of the test is $O(\mu(\sigma^{0.75}))$.

C. Conditions under UP

Under UP, monitors have no control over the paths between monitors, and the set of measurement paths $P$ is limited to the paths between monitors predetermined 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. To solve this issue, 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 will not be 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 $S_v := \{P_w : w \in N, w \neq v\}$ denotes the collection of path sets traversing non-monitors in $N \setminus \{v\}$. We use MSC($v$) to denote the minimum set
cover of $P_v$ by $S_v$, i.e., $MSC(v) := |V'|$ 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 $MSC(v) \leq \sigma - 1$. If $v$ is on a 2-hop path, then we define $MSC(v) := \sigma$.

**Theorem 13** ($k$-identifiability under UP). Network $G$ is $k$-identifiable under UP with measurement paths $P$: 

- **a)** if $MSC(v) > k$ for any non-monitor $v$;
- **b)** only if $MSC(v) > k - 1$ for any non-monitor $v$.

**Proof.** See [9].

**Testing algorithm:** The conditions in Theorem 13 provide an explicit way of testing the $k$-identifiability under UP, using tests of the form $MSC(v) > s$. 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 $MSC(v)$. The best-known algorithm with approximation guarantee is the greedy algorithm, which iteratively selects the set in $S_v$ 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 $GSC(v)$ denote the number of sets selected by the greedy algorithm. This immediately provides an upper bound: $MSC(v) \leq GSC(v)$. Moreover, since the greedy algorithm has an approximation ratio of $\log(|P_v|) + 1$ [11], we can also bound $MSC(v)$ from below: $MSC(v) \geq GSC(v)/(\log(|P_v|) + 1)$. Applying these bounds to Theorem 13 yields a pair of relaxed conditions:

- $G$ is $k$-identifiable under UP if $k < \left\lfloor \min_{v \in N} \frac{GSC(v)}{\log(|P_v|) + 1} \right\rfloor$;
- $G$ is not $k$-identifiable under UP if $k > \min_{v \in N} GSC(v)$.

These conditions can be tested by running the greedy algorithm for all non-monitors, each taking time $O(|P_v|^3 \sigma) = O(|P|^2 \sigma)$, and the overall test has a complexity of $O(|P|^3 \sigma^2)$ (or $O(\mu^3 \sigma^2)$ if there is a measurement path between each pair of monitors). However, we point out that it is unlikely that one can obtain stronger conditions based on Theorem 13 that are polynomial-time verifiable, as the greedy algorithm is known to give the best approximation for $MSC(v)$.

**IV. CHARACTERIZATION OF MAXIMUM IDENTIFIABILITY**

Although it is challenging to determine the exact value of the maximum identifiability $\Omega(G)$ without a (polynomial-time verifiable) necessary and sufficient condition for testing $k$-identifiability (it remains open as to whether it is NP-hard to determine the value of $\Omega(G)$), we will show that the conditions derived in Section III have a nice structure that allows us to provide tight upper and lower bounds on $\Omega(G)$.

**A. Maximum Identifiability under CAP**

Observing that both the sufficient and the necessary conditions in Corollary 6 are imposed on the same property, i.e., vertex-connectivity of the auxiliary graph $G^*$, we obtain a tight characterization of the maximum identifiability under CAP as follows. Here $\delta(G)$ is the vertex connectivity of $G$ defined in Definition 4.

**Theorem 14** (Maximum Identifiability under CAP). If $\delta(G^*) \leq \sigma - 1$, the maximum identifiability of $G$ under CAP, $\Omega^{CAP}(G)$, is bounded by $\delta(G^*) - 1 \leq \Omega^{CAP}(G) \leq \delta(G^*)$.

**Proof.** See [9].

**Remark:** In the special case of $\delta(G^*) = \sigma$ (note that $\delta(G^*) \leq \sigma$ by definition), $G^*$ must be a clique, which means that all non-monitors must be neighbors of monitors. By Corollary 7, this implies that $\Omega^{CAP}(G) = \sigma$.

**Evaluation algorithm:** Using the algorithm for determining network vertex connectivity in [10], we can compute $\delta(G^*)$ and evaluate $\Omega^{CAP}(G)$ by the bounds in Theorem 14 in $O(\sigma^{2.75})$ time. The special case of $\Omega^{CAP}(G) = \sigma$ can be checked separately in $O(\sigma)$ time using the condition in Corollary 7.

**B. Maximum Identifiability under CSP**

As in the case of CAP, we can leverage the analogy between the sufficient and the necessary conditions in Corollary 10 to bound the maximum identifiability under CSP from both sides. Specifically, let $\delta_{\text{min}} := \min_{m \in M} \delta(G_m)$ be the minimum vertex-connectivity for auxiliary graphs $G_m$. Then the maximum identifiability is bounded as follows.

**Theorem 15** (Maximum Identifiability under CSP). If $\min(\delta_{\text{min}}, \delta(G^*) - 1) \leq \sigma - 2$, the maximum identifiability of $G$ under CSP, $\Omega^{CSP}(G)$, is bounded by $\min(\delta_{\text{min}}, \delta(G^*) - 2) \leq \Omega^{CSP}(G) \leq \min(\delta_{\text{min}}, \delta(G^*) - 1)$.

**Proof.** See [9].

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

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

In particular, if $\delta(G^*) = 1$ (i.e., there is a cut-vertex in $G^*$), then $\Omega^{CSP}(G) = 0$, i.e., even single-node failures cannot always be localized.

The only cases when $\min(\delta_{\text{min}}, \delta(G^*) - 1) \leq \sigma - 2$ is violated are: (i) $\delta_{\text{min}} = \delta(G^*) = \sigma$, or (ii) $\delta_{\text{min}} = \sigma - 1$ and $\delta(G^*) = \sigma$. In case (i), $G_m$ is a clique for all $m \in M$, i.e., each non-monitor still has a monitor as a neighbor after removing $m$; by Corollary 11, this implies that $\Omega^{CSP}(G) = \sigma$. In case (ii), Corollary 10 (a) can still be applied to show that $\Omega^{CSP}(G) \geq \sigma - 2$, and one can verify that the condition in Corollary 11 is violated, which implies that $\Omega^{CSP}(G) \leq \sigma - 1$. In fact, we can leverage Corollary 12 to uniquely determine $\Omega^{CSP}(G)$ in this case. If condition (ii) in Corollary 12 is satisfied, then $\Omega^{CSP}(G) = \sigma - 1$; otherwise, $\Omega^{CSP}(G) = \sigma - 2$.

**Evaluation algorithm:** Evaluating $\Omega^{CSP}(G)$ by Theorem 15 involves computing the vertex-connectivities of the auxiliary graphs $G^*$ and $G_m$ ($\forall m \in M$) using the algorithm for determining network vertex connectivity in [10], which altogether takes $O(\mu |G|^3)$ time.

**C. Maximum Identifiability under UP**

Let $\Delta := \min_{v \in N} MSC(v)$ be the minimum set cover over all non-monitors. The conditions in Theorem 13 imply the following bounds on the maximum identifiability under UP.

**Theorem 16** (Maximum Identifiability under UP). The maximum identifiability of $G$ under UP, $\Omega^{UP}(G)$, with measurement paths $P$ is bounded by $\Delta - 1 \leq \Omega^{UP}(G) \leq \Delta$.

**Proof.** See [9].

**Remark:** Recall that $\Delta \leq \sigma$ by definition. In the special case of $\Delta = \sigma$, we know that all non-monitors are on 2-hop measurement paths, whose states can be determined independently. Thus, $\Omega^{UP}(G) = \sigma$ in this case.

**Evaluation algorithm:** The original bounds in Theorem 16 are hard to evaluate due to the NP-hardness of computing
MSC(·). As in Section III-C, we resort to the greedy algorithm, which implies the following relaxed bounds:

\[
\min_{v \in N} \frac{\text{GSC}(v)}{\log(|P_v|) + 1} - 1 \leq \Omega^{\text{UP}}(G) \leq \min_{v \in N} \text{GSC}(v). \tag{5}
\]

Evaluating these bounds involves invoking the greedy algorithm for each non-monitor, with an overall complexity of \(O(|P|^2 \sigma^2)\) (or \(O(\mu^4 \sigma^2)\) if all monitors can probe each other).

V. IMPACT OF PROBING MECHANISM

Given the above results, we are now ready to quantify the impact of the probing mechanism on node failure localization. We aim to quantify this impact by evaluating, using our bounds on the maximum identifiability, the number of simultaneous failures we can uniquely localize in a given network with a given monitor placement under each of the three probing mechanisms (CAP, CSP, UP). In this study, we assume (hop count-based) shortest path routing as the default routing protocol under UP, i.e., the measurement paths under UP are the shortest paths between monitors, with ties broken arbitrarily.

A. Topologies for Evaluation

We evaluate the proposed metrics on both synthetic and real network topologies detailed as follows.

1) Synthetic Topologies: We first consider synthetic topologies generated according to four widely used random graph models: Erdős-Rényi (ER) [12], Random Geometric (RG) graphs [13], Barabási-Albert (BA) graphs [14], and Random Power Law (RPL) graphs [15]. We randomly generate graph realizations of each model \(\mathcal{M}_i\), with each realization containing 20 nodes (i.e., \(|V| = 20\)). The generated graphs are then used to evaluate the impact of probing mechanisms. We now describe the models and present the corresponding network topologies.

2) Real Topologies: For real topologies, we use the Autonomous System (AS) topologies collected by the Rocketfuel [16] project, which represents IP-level connections between backbone/gateway routers of several ASes from major Internet Service Providers (ISPs) around the globe. The parameters of selected networks obtained from this project are listed in Table II, where we sort the networks according to their numbers of nodes.

B. Placement of Monitors

Since the maximum identifiability \(\Omega\) depends on the given placement of monitors, we want to randomize this given monitor placement for a comprehensive evaluation. To this end, we adopt an Enhanced Random Monitor Placement (ERMP) strategy, which consists of the following two steps.

\[
\text{ERMP} = \left\{ \begin{array}{ll}
\text{Step (i)} & \text{place monitors greedily to avoid the obvious cases of zero-maximum identifiability mentioned above;}
\text{Step (ii)} & \text{place additional monitors, if available, randomly.}
\end{array} \right.
\]

In this algorithm, it is assumed that the total number of monitors is sufficient for step (i) above.

C. Impact on Identifiability

1) Tightness of Bounds: To measure the impact of probing on the maximum identifiability \(\Omega\), we need tight bounds on \(\Omega\) under all probing mechanisms. Although we have achieved this theoretically by deriving upper and lower bounds that differ by at most one (Theorems 14, 15, 16), only the bounds under CAP and CSP can be evaluated efficiently, and the bounds under UP have to be relaxed by a logarithmic factor to be computable in polynomial time (see (5)). The first question is therefore how tight the relaxed bounds are.

To this end, we compare the original bounds (Theorem 16) and the relaxed bounds (5) on a variety of topologies synthetically generated from the models in Section V-A1 in two scenarios, i.e., sparsely-connected and densely-connected topologies. To make the models comparable in each scenario, we have tuned each model to generate graphs with the same number of nodes and (average) number of links. We evaluate both bounds on multiple graph instances per model, each with a fixed number of monitors placed by ERMP, and present the average lower/upper bounds in Fig. 3. As expected, the relaxed lower bounds are quite loose due to the logarithmic factor, but the relaxed upper bounds coincide with the original bounds for all graph instances. This indicates that although the relaxed upper bound \(\min_{v \in N} \text{GSC}(v)\) can be a logarithmic-factor larger than the original upper bound \(\Delta\) in the worst case, this worst case rarely occurs, and we can approximate \(\Delta\) by \(\min_{v \in N} \text{GSC}(v)\) to apply Theorem 16. This provides a tight characterization of \(\Omega^{\text{UP}}\) for large networks, where computing the original bounds is infeasible.

2) Comparison of Probing Mechanisms: We are now ready to compare \(\Omega^{\text{CAP}}\), \(\Omega^{\text{CSP}}\), and \(\Omega^{\text{UP}}\).

Comparison Using Random Topologies: Similar to Section V-C1, \(\Omega^{\text{CAP}}\), \(\Omega^{\text{CSP}}\), and \(\Omega^{\text{UP}}\) are compared on both sparsely-connected and densely-connected topologies generated from the four random graph models. Under each scenario, we generate multiple graph instances from each of the four models and sequentially place monitors in each instance using ERMP such that the set of monitors grows monotonically as the number of monitors increases. We then evaluate our bounds on the maximum identifiability \(\Omega^{\text{CAP}}\), \(\Omega^{\text{CSP}}\), and \(\Omega^{\text{UP}}\) for each graph instance under each monitor placement.

\[\text{In the case of } 0 \leq \Omega \leq 1, \text{ we use Definition 1 to uniquely determine the value of } \Omega, \text{ i.e., we test if (i) } P_v \neq \emptyset \text{ for any node } v \text{ and (ii) } P_v \neq P_w \text{ for any two nodes } v \text{ and } w.\]
the maximum identifiability, to quantify this capability as localize failed nodes from the health condition of end-to-end maximum ratio 100%. CAP can provide unique localization even if up to different probing mechanisms: while UP can barely localize a large differences in the maximum identifiabilities of the different 
monitors \( \Omega \) independently select 20 especially between \( \Omega \) maximum identifiability under different probing mechanisms, the results show clear differences between the topologies, the results show clear differences between the minimum fraction of non-monitors simultaneously fail, i.e., \( \Omega^{CF} = \sigma \), for all the graph models when \( \sigma = \lfloor V \rfloor / \sigma \). Note that \( \Omega \) eventually decreases as the number of monitors increases, as the maximum identifiability is always upper bounded by the total number of non-monitors \( \sigma = |V| - \mu \); we have verified that the normalized maximum identifiability \( \Omega / \sigma \) increases monotonically with \( \mu \).

Comparison Using AS Topologies: For AS topologies, we first compute the minimum number of monitors required by step (i) of ERMP, denoted by \( \mu_{c} \), and then set the number of monitors \( \mu \) as a fixed fraction of the total number of nodes \( |V| \) such that \( \mu \geq \mu_{c} \) for all the topologies. For each topology, we independently select 20 sets of monitors using ERMP (only \( \mu = \mu_{c} \) monitors in each set are randomly placed), under which \( \Omega^{CF} \), \( \Omega^{CF} \), and \( \Omega^{UP} \) are evaluated.

Table III shows bounds on the maximum identifiability averaged over different monitor placements for the Rocketfuel AS topologies. In Table III, AS1221 (Telstra) has the maximum ratio \( \mu_{c} / |V| \) of approximately 0.75. Thus, we set \( \mu / |V| \) to 0.8 in the evaluation. Similar to the case of random topologies, the results show clear differences between the maximum identifiability under different probing mechanisms, especially between \( \Omega^{CF} \) and the other two. For most of the networks, UP and CSP only guarantee unique localization of single-node failures, while CAP can handle multi-node failures for all the networks. Across the networks, we observe that the ordering of the normalized maximum identifiability \( \Omega / \sigma \) is roughly consistent with the ordering of the minimum fraction of monitors \( \mu_{c} / |V| \) required by ERMP: the larger \( \mu_{c} / |V| \), the smaller \( \Omega / \sigma \).

VI. CONCLUSION

We have studied the fundamental capability of a network to localize failed nodes from the health condition of end-to-end paths between monitors. We proposed a novel measure, called the maximum identifiability, to quantify this capability as the maximum number of simultaneous failures that can be uniquely localized. We studied this measure in detail for three representative families of probing mechanisms that offer different tradeoffs between the controllability of probes and the cost of implementation. For each family of probing mechanisms, we established necessary/sufficient conditions for unique failure localization based on the network topology, the placement of monitors, the constraints on measurement paths, and the maximum number of simultaneous failures. We further showed that these conditions lead to tight upper/lower bounds on the maximum identifiability that differ by at most one. 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 reveal that although incurring a higher implementation cost, giving the monitors more control over the routing of probes can significantly improve their capability to localize simultaneous failures.

REFERENCES


| Table III | Maximum Identifiability for Rocketfuel AS Topologies \((\mu/|V| = 0.8)\) |
|-----------|-----------------------------|
| AS        | \(\mu_c\) | \(\mu\) | \(\sigma\) | \(\Omega^{CF}\) | \(\Omega^{CF}\) | \(\Omega^{CF}\) |
| AS        | 1755      | 69     | 138    | 34         | [29.15, 29.3] | [6.6, 25] | 2.3         |
| AS        | 6461      | 121    | 146    | 36         | [8.3, 9.1]    | 1       | 1           |
| AS        | 3967      | 106    | 161    | 40         | [22.8, 23.5]  | [1.05, 1.1] | 0.75        |
| AS        | 3257      | 123    | 192    | 48         | [20, 20.6]    | 1       | 0.75        |
| AS        | 1221      | 240    | 255    | 63         | [17.45, 18.2] | 1       | 1           |
| AS        | 1239      | 360    | 484    | 120        | [31.1, 31.8]  | 1       | 0.7         |