Probe Station Selection for Robust Network Monitoring

Deepak Jeswani*‡, Nakul Korde*‡, Dinesh Patil*‡, Maitreya Natu‡, John Augustine†
‡Tata Research Development and Design Centre, Pune, India
†School of Physical and Mathematical Sciences, Nanyang Technological University, Singapore
Email: {jeswanideepak1986, nakulkorde,dineshpatilMIT}@gmail.com,
maitreya.natu@tcs.com, jea@ics.uci.edu

***Abstract*—Probing is a promising approach for network monitoring. An important problem that needs to be addressed while developing probing-based solutions is the selection of probe station nodes. Probe station nodes are the nodes that are instrumented with the functionality of sending probes and analyzing probe results. The placement of probe stations affects the diagnosis capability of the probes sent by the probe stations. The probe station placement also involves the overhead of instrumentation. Thus it is important to minimize the required number of probe stations without compromising on the required diagnosis capability of the probes. In this paper, we present a novel reduction of the Minimum Probe Station Selection problem to the Minimum Hitting Set problem. We show that the problem of probe station selection can be solved by using approximation algorithms for the Minimum Hitting Set problem.**

I. Introduction

The demand for monitoring the network for detection and localization of faults is becoming more and more critical with the increasing size and complexity of the network. In the past, various approaches have been proposed for network monitoring. One promising approach proposed in the past is based on probing [3][4]. Probing involves sending probes as test transactions in the network. The success and failure of these probes depend on the success and failure of the network components used by the probe. Probes such as pings and traceroutes can be used to check the network availability and latency. More sophisticated application-level probes can be used to test the application performance. Probing based techniques have various advantages over the traditional passive monitoring based techniques [10], such as (1) less instrumentation, (2) capability to compute end-to-end performance, (3) quicker localization, etc.

One of the biggest problems to address while developing probing-based monitoring solutions is *probe station selection*. The probe station selection problem addresses the problem of selecting nodes in the network where the probe stations should be placed. The probe stations are the nodes that send probes into the network and analyze probe results. The probe station nodes should be selected such that the required diagnosis capability can be achieved through probes. Furthermore, as the probe station selection involves an additional instrumentation cost, the number of probe stations need to be minimized.

In this paper, we address the problem of selecting probe stations in a network in order to localize node failures in the network. The proposed algorithms can be easily modified for other types of failures, such as link failures, application failures, etc. In order to simplify the problem, we limit the probe station selection for localizing at most k simultaneous node failures. The Minimum Probe Station Selection problem can be defined as:

> Given a network, find the minimal number of nodes in the network where the probe stations should be placed, such that any k node failures can be localized.

We show that the Minimum Probe Station Selection problem can be reduced to the Minimum Hitting Set problem. We propose to use the approximation algorithms for Minimum Hitting Set problem to intelligently place probe stations. The main contribution of this paper is a novel reduction of the Minimum Probe Station Selection problem to the Minimum Hitting Set problem. We validate the proposed approach through experimental evaluation.

The rest of the paper is organized as follows. We present related work in Section II. We describe the problem in more detail in Section III. We then present the reduction of Minimum Probe Station Selection problem to Minimum Hitting Set problem in Section IV. We present experimental evaluation in Section V followed by conclusion in Section VI.

II. Related work

The problem of probe station selection has been addressed in a variety of ways in the past. Authors in [6] and [11] propose to divide the network into clusters and strategically place probe stations (tracers) such that each cluster is near at least one tracer. Horton et al. [5] propose to place probe stations at high arity nodes and choose a routing policy to determine the direction of message forwarding. Authors in [7] propose to deploy a single network operations center and use explicitly routed probes along the network paths of interest. Authors in [1] manually choose probe stations segregated in four groups for different domains.

Most of the existing techniques (1) assume explicit routing of probes and (2) do not consider network failures. Explicit routing is not always viable in real-systems. Also, the probe station selection problem becomes more challenging when network failures are taken into account. In the past, some researchers have addressed the problem of probe station selection while solving the problem of diagnosing node and link failures [2], [9]. In this paper, we present a novel approach to address the Minimum Probe Station Selection problem by a systematic reduction of the Minimum Probe Station Selection problem to the Minimum Hitting Set problem. We show through experimental evaluation that the proposed algorithm outperforms the past algorithms.

III. Problem Description

For the ongoing discussion we assume static single-path routing. Thus packets between a particular source-destination pair always follow a single path that does not change with time. We refer to this routing model as the *consistent IP routing model*. The proposed approach can be extended to consider dynamic routing using a non-deterministic model.

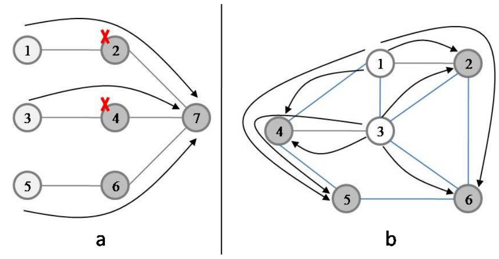

Fig. 1. (a) Three independent paths to node 7 allow detection of failure at node 7 even in the presence of two other node failures (nodes 2 and 4), Example topology with nodes 1 and 3 as probe stations. (b) Figure shows paths from probe station nodes 1 and 3 to other nodes. All nodes except node 5 have 2 independent paths from the probe stations, making node 5 the shadow node (assuming $k = 2$).

Addressing non-determinism in the network is part of our future work. A path from a probe station node to any other node is referred to as *probe path*. Two paths to the same destination are said to be *independent* if there are no common nodes in the path except the destination node, A node whose failure cannot be detected by the probe stations in a certain scenario of k node failures is termed as *shadow node*. For clarity we assume that the probe station nodes are fault tolerant and do not fail. However, it is easy to relax this assumption to use the proposed approach for detecting probe station failures as well.

The approach presented in this paper uses the following theorem:

Theorem 3.1: Assuming a consistent IP routing model with at most k failures in the network, a set of probe stations can localize any k non-probe-station node failures in the network if and only if there exist k independent probe paths to each non-probe-station node [9].

The example shown in Figure 1a explains the above theorem. Figure 1a shows paths to node 7 from the probe station nodes 1, 3, and 5. It can be seen that node 7 has 3 independent (node disjoint) paths from the probe station nodes 1, 3, and 5. Thus failure of node 7 can be detected even if there are two more failures in the network. For instance, if nodes 2 and 4 fail then probe station nodes 1 and 3 will not be able to detect the failure of node 7. With the assumption of at most 3 failures, the third independent path to node 7 from the probe station node 5 will not have any intermediate failures, making the probe station node 5 detect the failure

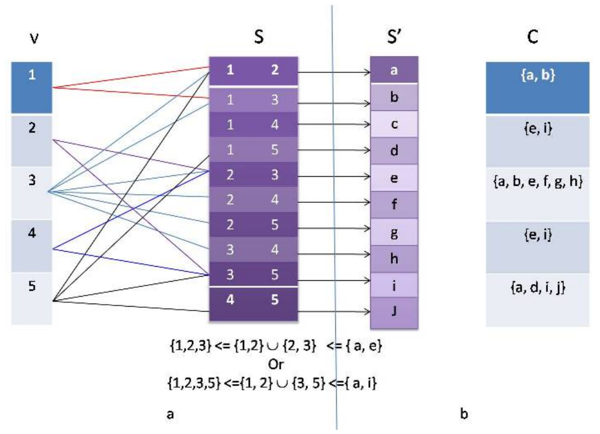

Fig. 2. Construction of Hitting Set problem instance from the Probe Station Selection problem instance

of node 7.

From the above theorem, the shadow nodes can be redefined as the nodes that have less than *k* independent paths from the probe stations. Figure 1b shows an example topology with nodes 1 and 3 as probe station nodes. We rely on the underlying routing model and do not demand an explicit routing of paths. Figure 1b shows the available paths from the probe station nodes to other nodes. It can be seen that nodes 2, 4, and 6 have 2 independent paths from the probe stations. However, with the given paths, node 5 does not have two independent paths from the probe station nodes. Paths from both the probe station nodes to node 5 have a common node 4. Assuming *k* equals 2, in the topology presented in Figure 1b, node 5 is the *shadow node*. The objective of the probe station selection algorithm is to select probe stations such that there are no shadow nodes in the network.

We next present a novel reduction of the Minimum Probe Station Selection problem to the Minimum Hitting Set problem. We show that the Minimum Probe Station Selection problem can be reduced to the Minimum Hitting Set problem in polynomial time. We then propose to use approximation algorithms for the Minimum Hitting Set problem to select the probe station nodes.

IV. Reduction of Minimum Probe Station Selection to Minimum Hitting Set

The Minimum Probe Station Selection problem has already been proved to be NP-Complete [9] using a reduction from the Set Cover problem. It can be easily proved that the log(n) inapproximability result of the Set Cover problem carries over to the Minimum Probe Station Selection problem as well. In this section, we reduce the Minimum Probe Station Selection problem to a dual of the Set Cover problem, namely, the Minimum Hitting Set problem. Using this reduction we show that a solution to the Minimum Hitting Set problem can provide a solution to the Minimum Probe Station Selection problem.

We model the network by an undirected graph $G(V, E)$, where the graph nodes, V, represent the network nodes (routers, end hosts) and the edges, E, represent the communication links connecting the nodes. We use $P_{u,v}$ to denote the path traversed by an IP packet from a source node u to a destination node v. We make the following assumptions about the underlying routing model: (1) the nodes use shortest paths to reach other nodes, (2) packets for the same destination are always forwarded to the same next hop by a node, (3) paths are symmetric, and (4) path between any two nodes is static and does not change with time. As a consequence, for a node s, the subgraph obtained by merging all the paths $P_{s,t}$ for every $t \in V$ has a tree topology. We refer to this tree for node s as its routing tree T_s. In what follows, we use the following definitions:

- Path nodes: We represent the nodes used on a path $P_{u,v}$ with a set $PN_{u,v}$
- Independent paths to a node: We define paths $P_{u_1,v}$ and $P_{u_2,v}$ to the same destination node v as independent if none of the nodes on $P_{u_1,v}$ is present on $P_{u_2,v}$ except the destination node v. That is $PN_{u_1,v} \cap PN_{u_2,v} = \{v\}$.

The Minimum Probe Station Selection problem is to select the smallest set of nodes as probe stations such that for every node v that is not a probe station, there are k independent paths from the probe stations to v. To facilitate the reduction, we define both the Minimum Probe Station Selection and Minimum Hitting Set problem precisely.

Probe Station Selection problem

> *Instance:* Graph $G(V, E)$, a routing tree T_u for each node $u \in V$, an integer k.
> *Problem:* Find the set $Q \subseteq V$ of least cardinality such that every node $u \in \{V - Q\}$ has k independent paths from the nodes in Q.

Hitting Set problem

Instance: Collection C of subsets of a finite set S.

Problem: Find the hitting set $H \subseteq S$ of least cardinality for C. A set $H \subseteq S$ is a hitting set of C if it contains at least one element from each subset in C.

We now provide a reduction from the Minimum Probe Station Selection to the Minimum Hitting Set such that a good solution for the Minimum Hitting Set problem (with a small hitting set size) will intuitively imply a good solution for the Minimum Probe Station Selection problem (with small number of probe stations).

The instance of the Minimum Hitting Set problem consisting of a finite set *S* and a collection C of subsets of S is constructed as follows. Given a Minimum Probe Station Selection instance (Graph *G(V,E)*, routing tree T_u for each node $u \in V$, integer *k*), create a Minimum Hitting Set instance as follows:

1) The elements of S are themselves sets. They are the $\binom{|V|}{k}$ distinct subsets, $S_1, \ldots, S_{\binom{|V|}{k}}$, of the set V, such that $|S_i| = k$.
2) Collection *C* of subsets of *S*: Each $C_i \in C$ (corresponding to vertex $v_i \in V$) is a subset of S and contains the elements $S_i \in S$ such that the set of nodes represented by S_i provide *k* independent paths to the node v_i.

It can be seen that the above explained reduction can be performed in polynomial time. For a network size of n nodes and diagnosis of at most k failures, the reduction can be performed in $O(n^k)$ operations. From the resulting hitting set $H \subseteq$ *S*, the probe station set *Q* can be derived as follows:

$$Q = \bigcup_{\forall H_i \in H} H_i \qquad (1)$$

The above reduction can be explained with the example shown in Figure 2. Figure 2a shows a set *V* of 5 nodes and a set *S* of $\binom{5}{2}$ combinations of the set *V*, where each combination $S_i \subseteq S$ is of size 2. A node $v \in V$ is connected to a node pair $S_i \in S$ if the set of nodes in S_i provide two independent paths to v. For instance, node 1 is connected to pairs *(1,2)* and *(1,3)*, as these pairs provide independent paths to node 1. For clarity the actual network and the paths from each node to every other node are not shown.

The set *S* is one-to-one mapped to a set S' in Figure 2b. For instance, the pair *(1,2)* and *(1,3)* in *S* are mapped to the elements *a* and *b* in the set S'. Based on the mapping of each node $v \in V$ to the node pairs in *S*, a collection *C* of subsets of S' is built. The collection *C* contains $|V|$ sets such that one node $v \in V$ corresponds to one set $C_i \in C$. If a node $v \in V$ is connected to *m* sets in the *S*, then the set C_i consists of corresponding *m* elements in the set S'. For instance, as node 1 is connected to pairs *(1,2)* and *(1,3)*, node *1* is mapped to the set {*a*,*b*} in *C*.

A Minimum Hitting Set solution for the instance in Figure 2b results in the solution *(a,e)*. The sets *a* and *e* in S' correspond to the sets *(1,2)* and *(1,3)* in *S*. Thus from the Minimum Hitting Set solution *(a,e)*, the probe station set solution *(1,2,3)* can be built. Note that another Minimum Hitting Set solution *(a,i)* results in the probe station set *(1,2,3,5)*. We propose to address this issue by assigning weights to the resulting hitting set solutions. The weight represents the number of nodes that get added to the resulting probe station set. A solution with minimal weight is preferred.

V. Experimental evaluation

In this section, we validate the proposed approach by presenting the experimental evaluation. We implemented a greedy approximation algorithm for the Minimum Hitting Set problem and used the reduction explained in Section IV to compute probe stations. For a problem instance with n sets, the computational complexity of the greedy approximation algorithm for computing hitting set is $O(n^2)$. Inapproximability results show that the greedy algorithm is essentially the best-possible polynomial time approximation algorithm for Minimum Hitting Set problem under plausible complexity assumptions.

We compare the results of the proposed approach with (1) Optimal algorithm based on combinatorial search, (2) Random node placement algorithm, (3) Max Degree algorithm [5], and (4) Shadow Node Reduction (SNR) algorithm [9].

We apply the algorithms to build a probe station set that can localize at most 3 node failures. We were unable to run the optimal algorithm on larger networks because of its combinatorial nature and large execution time. Hence for evaluation of the proposed algorithms with the optimal algorithm, we have conducted experiments on network size from 10 to 50 nodes with average node degree 4

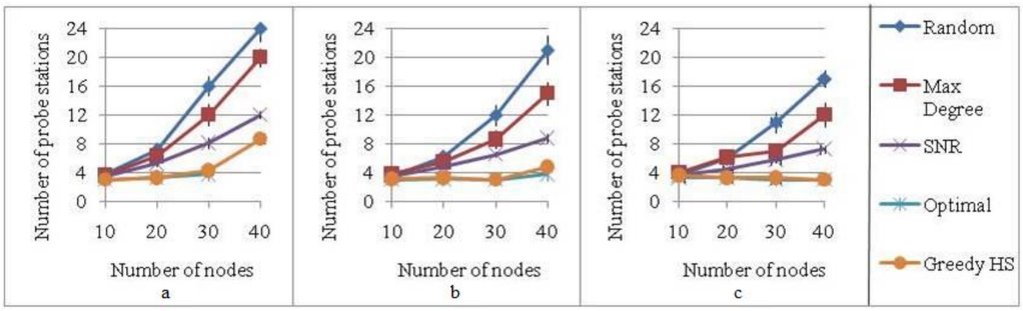

Fig. 3. Number of probe stations computed by different algorithms for different network sizes and avg. node degree (a) =4, (b) =5, (c) =6.

to 6. We have used BRITE [8] to generate network topologies. Each point plotted on the graph is an average of 20 runs. We have plotted the 95% confidence intervals.

Figure 3 shows the number of probe stations computed by various algorithms. It can be seen that the number of probe stations computed by the Hitting Set based algorithm is close to optimal. Also note that the SNR algorithm [9] does not perform as good as the Hitting Set based algorithm. The Max Degree and the Random node placement algorithms compute probe station sets of significantly larger size.

Figure 3 also shows the results for different values of average node degree. It can be seen that fewer number of probe stations are required with increasing average node degree. With higher average node degree, the nodes are more densely connected, providing more paths from the probe stations to the nodes. Thus with higher average node degree fewer number of probe stations are able to provide more independent paths.

VI. Conclusion and Future Work

We addressed the problem of selecting probe stations in a network in order to monitor the network for node failures. We presented a novel reduction of the Minimum Probe Station Selection problem to the Minimum Hitting Set problem. We presented experimental evaluation of the proposed approach and demonstrated that the proposed algorithm outperforms past algorithms.

As part of our ongoing work, we are working on further decreasing the computational complexity of the algorithm to make it usable for larger values of k. The algorithms in this paper ensure the availability of k independent probe paths but do not aim to optimize probe traffic or the localization time. An interesting approach to pursue is to identify suitable probes based on the criteria of minimizing probe traffic or localization time, and then attempt to minimize the number of probe stations. The path attributes such as latency, loss rate, bandwidth, etc. can also be considered in this approach.

References

[1] Cooperative association for Internet data analysis (CAIDA). the skitter project. Technical report, http://www.caida.org/tools/measurement/skitter/index.html, 2001.

[2] Y. Bejerano and Rajeev Rastogi. Robust monitoring of link delays and faults in IP networks. In *IEEE INFOCOM, San Francisco, CA*, Mar 2003.

[3] M. Brodie, I. Rish, and S. Ma. Optimizing probe selection for fault localization. In *Distributed Systems Operations Management*, pages 1147–1157, 2001.

[4] M. Brodie, I. Rish, S. Ma, G. Grabarnik, and N. Odintsova. Active probing. Technical report, IBM, 2002.

[5] J. D. Horton and A. Lopez-Ortiz. On the number of distributed measurement points for network tomography. In *Internet Measurement Conference, IMC*, 2003.

[6] S. Jamin, C. Jin, Y. Jin, Y. Raz, Y. Shavitt, and L. Zhang. On the placement of Internet instrumentation. In *IEEE INFOCOM, Tel Aviv, Israel*, Mar 2000.

[7] F. Li and M. Thottan. End-to-end service quality measurement using source-routed probes. In *25th Annual IEEE Conference on Computer Communications (INFOCOM), Barcelona, Spain*, Apr 2006.

[8] A. Medina, A. Lakhina, I. Matta, and John Byers. Brite: An approach to universal topology generation. In *International Workshop on Modeling, Analysis and Simulation of Computer and Telecommunications Systems-MASCOTS'01, Cincinnati, Ohio*, Aug. 2001.

[9] M. Natu and A. S. Sethi. Probe station placement for robust monitoring of networks. *Journal of Network and Systems Management*, 2007.

[10] M. Stemm, R. Kaltz, and S. Seshan. A network measurement architecture for adaptive applications. In *IEEE INFOCOM*, Mar 2000.

[11] W. Theilman and K. Rothermel. Dynamic distance maps of the Internet. In *IEEE INFOCOM'2000*, Mar 2000.