Zone Recovery Methodology for Probe-Subset Selection in End-to-end Network Monitoring

H. Cenk Ozmutlu

hco@uludag.edu.tr
Dept. of Industrial Engineering
School of Engineering and Architecture
Uludag University
Gorukle, Bursa, 16059, Turkey

Natarajan Gautam, Russell Barton

ngautam@psu.edu, rbarton@psu.edu
Dept. of Industrial and Manufacturing Engineering
Pennsylvania State University
University Park, PA, 16802, USA

ABSTRACT

To predict the delay between a source and a destination as well as to identify anomalies in a network, it is possible to continuously monitor the network by sending probes between all sources and destinations. However, it is of prime importance to keep the number of probes to a minimum and yet be able to reasonably predict the delays and identify anomalies. In this paper we state and solve a mathematical programming problem, namely the Zone Recovery Methodology (ZRM), to optimally select a subset of ping-like probes to monitor networks where the topology and routing information are not known. A polynomial-time heuristic is developed. The application of ZRM on randomly generated topologies yielded 73.55% reduction in the number of monitored paths on average. In other words, networks can be successfully monitored using only 26.45% of the available probes. Moreover, the performance of ZRM increases (percentage of the monitored paths decreases) as the size of the topology increases.

Keywords: Network management, quality of service, monitoring, end-to-end delay, anomaly detection

1. INTRODUCTION

Internet's utilization rate has increased tremendously as the users and applications have grown at an astonishing rate. The increasing popularity of web-based applications is the most significant contributor to the current congestion of the Internet [3], [12]. The problems due to network performance have led users to demand specific Quality-of-Service (QoS) levels for their applications. In this research, we concentrate on developing an end-to-end network monitoring strategy to obtain real-time information on network delay and anomalies in order to provide QoS to the users.

Predictions about end-to-end delay and location of anomalies or "hot-spots" can be done by continuously monitoring the network [13]. This monitoring strategy requires sending probes between all sources and destinations in the network. Such a strategy with large number of probes would not only lead to excessively large files to store and browse through in real time but also significant traffic due to the probes themselves. The main aim of this paper is to develop a monitoring strategy that uses an optimal subset of probes that is both small enough to handle and not contribute to the congestion, as well as large enough to make accurate predictions [14].

The scenario considered in this paper is one where there are network nodes around a "cloud" [15] and the objective is to probe and obtain the end-to-end delay between any two nodes around the cloud as well as any anomalies within the cloud. It is crucial to note that the network topology and routes within the cloud are unknown. In an earlier paper [8], the authors considered a scenario where the topology and the routes of all paths are known. The problem formulation, methodology and monitoring strategy are significantly different if the topology and routes are unknown. Our approach, called the Zone Recovery Methodology (ZRM), can be formulated as a non-linear integer program, which can be solved in exponential time. In this paper we develop a heuristic for ZRM, which is a polynomial-time algorithm.

In Section 2, we describe the objective of this study, analyze the model and state the assumptions. In Section 3, we describe the mathematical programming formulation. In Section 4, we develop a heuristic to solve the problem and present numerical examples. In Section 5, we state the conclusions of this work.

2. PROBLEM DEFINITION AND MODEL ANALYSIS

Consider a network cloud with *N* nodes that access the cloud. An example with *N = 3* is shown in Figure 1. The traditional monitoring strategy is to send probes from each of the *N* nodes to the remaining *N-1* nodes, resulting in a total of *N(N-1)* probes. The objective of this study is to select an optimal subset of these *N(N-1)* probes. The data obtained from the probes sent between any Source-Destination (S-D) pair among the *N* nodes can be used to calculate the one-way delays as well as locate anomalies in the cloud. A critical assumption is that the topology and routing within a cloud is unknown. Further, we assume that traceroutes are either blocked or produce no useful information for topology and routing within the cloud. However in this research, we use unidirectional ping-like probes[**] (which will be referred to as ping probes) that provide only end-to-end delays, similar to the datasets collected by tools such as SURVEYOR at the Advanced Network and Services, Inc. [4].

The monitoring strategy developed in this paper called the Zone Recovery Methodology (ZRM) consists of two phases: the initial phase to determine the optimal subset of probes and the final phase that uses the optimal subset for monitoring. In the initial phase, all available paths are monitored for a certain time period (which will be referred as the initial sampling period). The delay values of all paths (between sources and destinations) are used to identify the relationship between anomalies and paths. After the relationship between anomalies and paths are recovered, ZRM selects the subset of probes based on the occurrences of simultaneous packet losses or high delays on various paths. In the final phase, ping probes are sent between selected end nodes. Using the information from the probes, end-to-end delay between all nodes and anomalies within the cloud can be inferred.

The main idea behind ZRM is that anomalies occurring at different locations will affect different sets of paths. Each unique set of paths with high delays will correspond to a unique *zone* in the topology. A zone may consist of a single arc or multiple adjacent arcs and nodes (the necessity for the zone definition is demonstrated later in this section). Using end-to-end delay data and common occurrences of high delays on paths, the zones in the topology are recovered. After identifying the zones, a subset of the paths is selected that can identify each zone uniquely. ZRM depends on the following assumptions:

1. The initial sampling period is long enough to contain all possible anomalies,
2. There is at most one anomaly per sampling and monitoring time unit,
3. Each arc is covered by at least two probes.

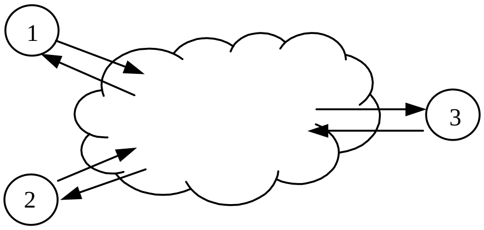


Figure 1: The cloud formation of Example 1

We now demonstrate the ZRM approach (see Example 1 in Figure 1). Although ZRM assumes that the topology and routing information is unknown, the topology and routing information is provided in Figure 2 to explain how ZRM works. Since ZRM assumes that the topology between nodes 1, 2 and 3 is unknown, we can only use the delay values of the probes that are sent between nodes 1, 2 and 3.

[**] Unix version of ping (ping -s "destination")

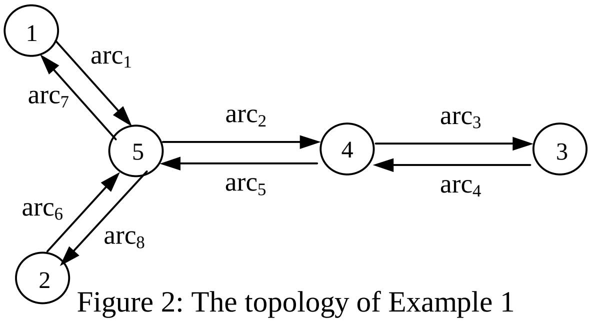


Figure 2: The topology of Example 1

Figure 3 illustrates the routes of the unidirectional probes that will be used to monitor the topology. Notice that opposite directions of an arc between two nodes are represented with two separate arcs. Consequently, Example 1 includes six probes, which are sent from *node*$_1$ to *node*$_2$ (*path*$_1$), from *node*$_1$ to *node*$_3$ (*path*$_2$), from *node*$_2$ to *node*$_1$ (*path*$_3$), from *node*$_2$ to *node*$_3$ (*path*$_4$), from *node*$_3$ to *node*$_1$ (*path*$_5$), and from *node*$_3$ to *node*$_2$ (*path*$_6$).

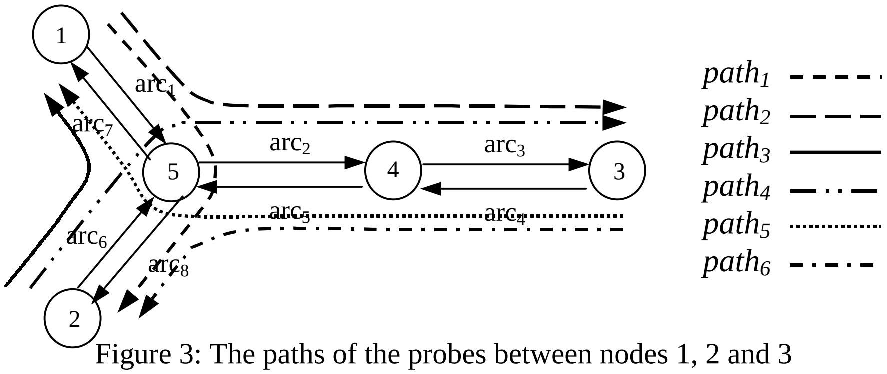


Figure 3: The paths of the probes between nodes 1, 2 and 3

Anomalies in the network can be detected only by simultaneous high delays or packet losses (which can be registered as abnormally high delays). Hence, anomalies will be expressed in terms of the level of delay values of the observed paths. Any delay value over a preset limit will be considered as an anomaly along the path of the corresponding probe. In Example 1, anomalies in each arc affect different sets of probes in terms of high delays. Table 1 demonstrates the possible anomaly locations, and the corresponding sets of paths affected by the anomalies. Note that the ping probes will provide data only to construct the second column of Table 1. Due to the assumption that there can be only one anomaly in the system during a single sampling period, each unique pattern in the second column of Table 1 corresponds to a unique zone. For Example 1, it is impossible to observe any other combination of affected paths (such as *path*$_1$, *path*$_2$ and *path*$_3$) with a single anomaly.

Table 1: Possible anomaly locations and the corresponding set of affected paths

Location of anomaly	**Set of effected paths**
arc_1	$path_1$, $path_2$
arc_2	$path_1$, $path_4$
arc_3	$path_1$, $path_4$
arc_4	$path_5$, $path_6$
arc_5	$path_5$, $path_6$
arc_6	$path_3$, $path_4$
arc_7	$path_3$, $path_5$
arc_8	$path_1$, $path_6$

Moreover, it is impossible to differentiate between the anomalies occurring at arc_2 and arc_3 (also between arc_4 and arc_5), since they have the same set of affected paths. Consequently, arc_2 and arc_3 are considered as a single zone. However, this is not a major concern, since a zone with multiple arcs can be treated as a single arc. All the paths that use a zone must enter from the initial node and leave at the end node of the zone. The paths cannot enter (leave) at an intermediate node in a zone. Otherwise, there will be anomalies on both sides of the intermediate node where the path enters (leaves) the zone, and this path can be used to identify at which part of the zone the anomaly occurred. As a result, both parts of the zone will be declared as unique zones. Occurrences of zones with multiple arcs and nodes are demonstrated in Figure 3. For example, every path traversing arc_2 also traverses arc_3, and there are no paths entering or exiting at node 4. In other words, arc_2 and arc_3 act as a single arc. Unique combinations of high delay patterns and their corresponding zones are provided in Table 2. Once the relationships between the zones and high delay values are identified, they can be used to trace the location of anomalies during the final phase, i.e. the monitoring stage. For example, observing the delay pattern "$path_1$ and $path_2$ are high, and other paths are low" will imply that there is an anomaly in $zone_1$.

Table 2: Unique combinations of high delay patterns and their corresponding zones

Unique set of effected paths	**Corresponding zone**	**Location of the anomaly**
$path_1$, $path_2$	$zone_1$	arc_1
$path_1$, $path_4$	$zone_2$	arc_2 or arc_3
$path_5$, $path_6$	$zone_3$	arc_4 or arc_5
$path_3$, $path_4$	$zone_4$	arc_6
$path_3$, $path_5$	$zone_5$	arc_7
$path_2$, $path_6$	$zone_6$	arc_8

We have demonstrated that it is possible to identify zones in a topology using end-to-end delay values obtained from ping probes. Although we cannot reduce the number of probes to be monitored in Example 1, there are other cases (see examples in Sections 4.2-4.5) where it is possible to identify zones without observing all the probes. In the following sections, we provide a mathematical programming formulation and a polynomial-time heuristic for the probe-subset selection based on ZRM.

3. THE GRAPH THEORY FORMULATION FOR ZRM

Given the delay data obtained by the ping probes between end-to-end S-D pairs during the initial sampling period, ZRM (explained below) selects a subset of end-to-end S-D pairs with paths that uniquely identify the zones of the topology, while minimizing the size of the subset. The relationship between zones and the probes are developed based on the simultaneous high delay values of the probes. After identifying the zones and their corresponding paths, ZRM aims to cover each zone with at least two paths that can uniquely identify the zone. Note that ZRM has predefined relations between paths (probes) and zones, hence differs from the classical node/arc covering problem [1][7]. ZRM can be stated as a mathematical programming problem, more precisely a binary Integer Non-Linear Programming (INLP) formulation as follows:

$$\min \qquad z = \sum_{i=1}^{n\_paths} path_i$$

$$s.t..$$

$$\prod_{i=1}^{(n\_paths-1)} \prod_{j=i+1}^{n\_paths} (a_{k,i} * a_{k,j} * p_{i,j} * path_i * path_j - 1) = 0 \qquad k = 1....M$$

$$path_i = 0 \; or \; 1 \quad i = 1....n\_paths$$

where

N = number of end nodes (intermediate nodes are unknown)

M = number of zones

n_paths = number of unidirectional end-to-end paths
= $N*(N-1)$

$$path_j = \begin{cases} 1 & \text{if } path_j \text{ is included in the probe subset, } j = 1..n\_paths \\ 0 & \text{otherwise} \end{cases}$$

A = the *zone-path* matrix
= $[a_{k,i}]$, $k = 1,\ldots,M$, $i = 1,\ldots, n\_paths$

$$a_{k,i} = \begin{cases} 1 & \text{if } zone_k \in path_i \\ 0 & \text{otherwise} \end{cases}$$

$p_{i,j}$ = inner product of columns that correspond to $path_i$ and $path_j$

$$= (a_i)^T (a_j) = \sum_{k=1}^{M} a_{k,i} * a_{k,j}$$

In the mathematical formulation of ZRM, there is one binary variable for each path, and the number of constraints is equal to the number of zones. In order to be eligible to identify $zone_k$, $a_{k,i}$ and $a_{k,j}$ should be equal to one, in other words both $path_i$ and $path_j$ should traverse $zone_k$. However, $path_i$ and $path_j$ are eligible to identify $zone_k$ if $zone_k$ is the only common zone in $path_i$ and $path_j$. When $zone_k$ is the only common zone in $path_i$ and $path_j$ then $p_{i,j}$ is equal to one. Otherwise, $p_{i,j}$ is either equal to 0 (i.e. no common zone in $path_i$ and $path_j$) or greater than 1 (i.e. there are more than one common zone in $path_i$ and $path_j$). Therefore $path_i$ and $path_j$ are eligible to uniquely identify $zone_k$ when $a_{k,i} = a_{k,i} = p_{i,j} = 1$. If $path_i$ and $path_j$ are selected ($path_i = path_j = 1$) and if $path_i$ and $path_j$ are eligible to identify $zone_k$ ($a_{k,i} = a_{k,i} = p_{i,j} = 1$), only then ($a_{k,i} * a_{k,i} * p_{i,j} * path_i * path_j - 1$) is equal to zero. Since ($a_{k,i} * a_{k,i} * p_{i,j} * path_i * path_j - 1$) is equal to zero, the constraint for $zone_k$ is satisfied irrespective of the value of the remaining multipliers. Formulation 1 selects the optimal subset of paths that will uniquely identify each zone, while minimizing the number of paths (since the objective is to minimize the number of paths).

4. IMPLEMENTATION OF ZRM

In the mathematical formulation, the number of variables is equal to n_paths, and there are $N*(N-1)$ unidirectional paths ($n\_paths = N*(N-1)$). Since all the variables are binary, there are at most $2^{N*(N-1)/2}$ alternative solutions, making the worst case complexity of a complete enumeration algorithm for the mathematical formulation of the order of $O(2^{n^2})$. Since the worst-case scenario of the mathematical formulation is in exponential time, we have developed an alternative polynomial-time heuristic for ZRM, which is explained in the next section. To demonstrate the effectiveness of ZRM, the polynomial-time heuristic for ZRM is applied on a real network topology, namely the vBNS backbone topology.

4.1. A Polynomial-Time Heuristic for ZRM (HZRM)

HZRM is a polynomial-time heuristic for ZRM. Every iteration of HZRM selects a pair of paths to uniquely identify a particular zone. During the path subset selection process, the paths selected at previous iterations of the algorithm are given priority over paths that are never used, since paths can be used to identify multiple zones, and since using previously selected paths does not contribute additional cost (in other words, using previously selected paths does not increase the number of probes in the selected subset). At any given point, if any of the selected paths is not eligible to identify any unidentified zones, then a greedy selection is made: the path that covers the most unidentified zones is selected, since it is eligible to identify the most number of unidentified zones. The heuristic terminates when all the zones are uniquely identified by a pair of paths. The formal algorithm of HZRM is given after the notation below:

L_final = the set of selected paths for monitoring
L_active = the set of selected paths with unidentified zones
L_passive = the set of paths that are not selected

Z_active = the set of unidentified zones

$Z\_current$ = the set of unidentified zones of the path under consideration

S_j = number of unidentified zones covered by $path_j$

A = the *zone-path* matrix

= $[a_{i,j}]$, $i = 1, \ldots, M$, $j = 1, \ldots, n\_paths$

$$a_{i,j} = \begin{cases} 1 & \text{if } zone_i \in path_j \\ 0 & \text{otherwise} \end{cases}$$

Step 1: Set $L\_final = L\_active = Z\_current = \varnothing$, and $Z\_active = \{zone_i, i = 1, \ldots, M\}$. $L\_passive = \{path_j, j = 1, \ldots, n\_paths\}$

Step 2: For each $path_j$, $j = 1, \ldots, n\_paths$, calculate S_j, number of unidentified zones covered by $path_j$. At this stage, S_j is equal to the column sum A.

$$S_j = \sum_{i=1}^{M} a_{i,j}$$

Step 3: Choose the largest nonzero S_j, such that $path_j \in L\_passive$ (ties are broken arbitrarily). Note that this step can be reached only when $L\_active = \varnothing$, consequently there is no need to look for a path in L_active. Let $S_k = \{\max(S_j) \mid path_j \in L\_passive\}$.

If $S_k = 0$, then terminate. The paths in L_final will constitute the subset of the S-D pairs to be monitored.

Else $L\_final = L\_final \cup \{path_k\}$,

$L\_active = L\_active \cup \{path_k\}$, the $path_k$ is attached to the end of L_active

Step 4: At this step a path is selected, and it is used in the following steps. We call this path the current path, which will be referred as $path_m$ in the remaining of the algorithm.

If $L\_active = \varnothing$, then go to Step 3.

Else Let $path_m$ be the first path in L_active then

$Z\_current = \{zone_j \mid zone_j \in path_m\} \cap Z\_active$, and

$L\_active = L\_active - \{path_m\}$.

Step 5: The unidentified zones of $path_m$ are determined and identified in the following steps.

If $Z\_current = \varnothing$, then go to Step 4.

Else Let $zone_c$ be the first zone in $Z\_current$

$Z\_current = Z\_current - \{zone_c\}$

Step 6: Find a path (which we will refer as $path_k$) in L_active such that:

i) $a_{c,k} = 1$ (to ensure $zone_c$ is a part of $path_k$),

ii) $(a_k)^T (a_m) = \sum_{i=1}^{M} a_{i,k} * a_{i,m} = 1$ (to ensure $path_k$ and $path_m$ have only one common zone)

If such a path ($path_k$) exists then

i) Register the pair $path_k$ and $path_m$ for identifying $zone_c$.

ii) Remove the identified zone from the unidentified zone set. $Z\_active = Z\_active - \{zone_c\}$

iii) Update S_j values (only the paths that include $zone_c$ are required to be updated):

If $a_{c,j} = 1$ then $S_j = S_j - 1$, $j = 1, \ldots, n\_paths$

iv) Go to Step 5.

Else proceed to the next step.

Step 7: Find a path (which we will refer as $path_k$) in *L_passive* such that:

i) $a_{c,k} = 1$ (to ensure $zone_c$ is a part of $path_k$),

ii) $(a_k)^T(a_m) = \sum_{i=1}^{M} a_{i,k} * a_{i,m} = 1$ (to ensure $path_k$ and $path_m$ have only one common zone)

If such path ($path_k$) exists then

- i) Register the pair $path_k$ and $path_m$ for identifying $zone_c$.
- ii) Remove the identified zone from the unidentified zone set: *Z_active* = *Z_active* - {$zone_c$}
- iii) Update S_j values (only the paths that include $zone_c$ are required to be updated):
  If $a_{c,j} = 1$ then $S_j = S_j - 1$, $j = 1, \ldots, n\_paths$
- iv) *L_passive* = *L_passive* - {$path_k$}
- v) *L_active* = *L_active* ∪ {$path_k$}
- vi) *L_final* = *L_final* ∪ {$path_k$}
- vii) Go to Step 5.

Else Go to Step 5.

4.2. Demonstration of HZRM

We demonstrate a few steps in solving Example 1 of Section 2 to provide some insight on how HZRM operates. From the data provided in Table 2, we have constructed the *zone-path* matrix, *A*. Notice that every row in Table 2 corresponds to a row in *A* (e.g. the first row in Table 2 corresponds to $zone_1$, and an anomaly in $zone_1$ affects $path_1$ and $path_2$, therefore the only nonzero values in the first row of *A* are in the columns of $path_1$ and $path_2$).

$$A = [a_{i,j}] = \begin{array}{cccccc} path_1 & path_2 & path_3 & path_4 & path_5 & path_6 \\ \end{array}$$
$$\left(\begin{array}{cccccc} 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 \end{array}\right)$$

Step 1: *L_final* = *L_active* = *Z_current* = ∅, and *Z_active* = {$zone_1$, $zone_2$, $zone_3$, $zone_4$, $zone_5$, $zone_6$}. *L_passive* = {$path_1$, $path_2$, $path_3$, $path_4$, $path_5$, $path_6$}

Step 2: Calculate S_j's, j=1,…,6

L_passive

		$path_1$	$path_2$	$path_3$	$path_4$	$path_5$	$path_6$
Z_passive	$zone_1$	1	1	0	0	0	0
	$zone_2$	1	0	0	1	0	0
	$zone_3$	0	0	0	0	1	1
	$zone_4$	0	0	1	1	0	0
	$zone_5$	0	0	1	0	1	0
	$zone_6$	0	1	0	0	0	1
	S_j	2	2	2	2	2	2

Step 3: S_1 = max (S_j), j=1,…,6. $S_2 > 0$, *L_final* = {$path_1$}, *L_active* = {$path_1$}

		L_active	*L_passive*				
		$path_1$	$path_2$	$path_3$	$path_4$	$path_5$	$path_6$
Z_passive	$zone_1$	1	1	0	0	0	0
	$zone_2$	1	0	0	1	0	0
	$zone_3$	0	0	0	0	1	1
	$zone_4$	0	0	1	1	0	0
	$zone_5$	0	0	1	0	1	0
	$zone_6$	0	1	0	0	0	1
	S_j	2	2	2	2	2	2

Step 4: *L_active* ≠ ∅, $path_1$ is the first path in *L_active* and *Z_current* = {$zone_1$, $zone_2$}, *L_active* = *L_active* - {$path_1$} = ∅.

Step 5: *Z_current* ≠ ∅. $zone_1$ is the first zone in *Z_current*. *Z_current* = *Z_current* - {$zone_1$} = {$zone_2$}

Step 6: *L_active* = ∅, there is no path that fits the requirements

Step 7: If we look at the row of $zone_1$, the only other "1" is at the second column, and it satisfies the conditions of Step 7 ($a_{1,2} = 1$, $(a_1)^T*(a_2) = 1$). Register the pair ($path_1$, $path_2$) as the identifier of $zone_1$. Update S_j, j=1,…,6, values (see the matrix below), *L_passive* = *L_passive* - {$path_2$} = {$path_3$, $path_4$, $path_5$, $path_6$}, *L_active* = *L_active* ∪ {$path_2$} = {$path_2$}, *L_final* = *L_final* ∪ {$path_2$} = {$path_1$, $path_2$}. Go to Step 5.

			L_active				
		$path_1$	$path_2$	$path_3$	$path_4$	$path_5$	$path_6$
	$zone_1$	1	1	0	0	0	0
Z_passive	$zone_2$	1	0	0	1	0	0
	$zone_3$	0	0	0	0	1	1
	$zone_4$	0	0	1	1	0	0
	$zone_5$	0	0	1	0	1	0
	$zone_6$	0	1	0	0	0	1
	S_j	1	1	2	2	2	2

Step 5: *Z_current* ≠ ∅. $zone_2$ is the first zone in *Z_current*. *Z_current* = *Z_current* - {$zone_2$} = ∅

Step 6: There is no path that fits the requirements in *L_active*

Step 7: If we look at the row of $zone_2$, the only other "1" is at the fourth column, and it satisfies the conditions of Step 7 ($a_{2,4} = 1$, $(a_1)^T*(a_4) = 1$). Register the pair ($path_1$, $path_4$) as the identifier of $zone_2$. Update S_j, j=1,…,6, values (see the matrix below), *L_passive* = *L_passive* - {$path_4$} = {$path_3$, $path_5$, $path_6$}, *L_active*

= *L_active* ∪ {$path_4$} = {$path_2$, $path_4$}, *L_final* = *L_final* ∪ {$path_4$} = {$path_1$, $path_2$, $path_4$}. Go to Step 5. Notice that we have rearranged the columns of the matrix for improved visual demonstration.

			L_active		*L_passive*		
		$path_1$	$path_2$	$path_4$	$path_3$	$path_5$	$path_6$
	$zone_1$	1	1	0	0	0	0
	$zone_2$	1	0	1	0	0	0
Z_passive	$zone_3$	0	0	0	0	1	1
	$zone_4$	0	0	1	1	0	0
	$zone_5$	0	0	0	1	1	0
	$zone_6$	0	1	0	0	0	1
	S_j	0	1	1	2	2	2

Step 5: *Z_current* = ∅. Goto Step 4.

Step 4: *L_active* ≠ ∅, $path_2$ is the first path in *L_active* and *Z_current* = {$zone_6$}, *L_active* = *L_active* - {$path_2$} = {$path_4$}.

Step 5: *Z_current* ≠ ∅. $zone_6$ is the first zone in *Z_current*. *Z_current* = *Z_current* - {$zone_6$} = ∅.

Step 6: There is no path that fits the requirements in *L_active*

Step 7: If we look at the row of $zone_2$, the only other "1" is at the column corresponding to $path_6$, and it satisfies the conditions of Step 7 ($a_{6,6} = 1$, $(a_2)^T*(a_6) = 1$). Register the pair ($path_2$, $path_6$) as the identifier of $zone_6$. Update S_j, j=1,…,6, values (see the matrix below), *L_passive* = *L_passive* - {$path_6$} = {$path_3$, $path_5$}, *L_active* = *L_active* ∪ {$path_6$} = {$path_4$, $path_6$}, *L_final* = *L_final* ∪ {$path_6$} = {$path_1$, $path_2$, $path_4$, $path_6$}. Go to Step 5. Notice that we have rearranged the columns and rows of the matrix for clarity.

				L_active		*L_passive*	
		$path_1$	$path_2$	$path_4$	$path_6$	$path_3$	$path_5$
	$zone_1$	1	1	0	0	0	0
	$zone_2$	1	0	1	0	0	0
	$zone_6$	0	1	0	1	0	0
Z_passive	$zone_3$	0	0	0	1	0	1
	$zone_4$	0	0	1	0	1	0
	$zone_5$	0	0	0	0	1	1
	S_j	0	0	1	1	2	2

At this point, HZRM uniquely identified $zone_1$, $zone_2$ and $zone_6$, with the pairs ($path_1$, $path_2$), ($path_1$, $path_4$) and ($path_2$, $path_6$), respectively. The remaining steps of HZRM are similar to the demonstration given above, therefore we will not provide the remaining steps of the solution for Example 1. The solution of the heuristic is the same as that of the optimal non-linear integer programming formulation. We begin the performance evaluation of HZRM by comparing HZRM's complexity to that of the non-linear integer programming formulation. The complexity of the non-linear integer programming formulation is given at the beginning of Section 4. The computational complexity of HZRM is calculated as follows.

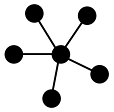
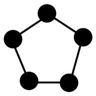
4.3. Computational Complexity of HZRM

At each iteration of the heuristic, a single zone is uniquely identified. For each zone, two paths are picked (either from previously selected or new paths). For selection of each path, the maximum number of paths that can be considered is *n_paths* (*n_paths* calculations for each path selection). In addition, after each zone selection process, S_j, $j = 1,\ldots,$ *n_paths*, are updated (all together there are 3**n_paths* iterations for each zone identification). The path selection is performed until every zone is identified (M times). Overall there are 3* *n_paths*M* calculations. Since *n_paths* is equal to $N*(N-1)$, the worst-case computational effort for the heuristic is $3*N*(N-1)*M$, and the complexity of the heuristic is in the order of $O(N^2M)$. Hence HZRM is a polynomial-time algorithm. Next, we compare the results of HZRM with the optimal solutions on common network structures for the cloud, namely a star topology, a loop (or a cycle) topology and a linear topology.

4.4. Comparison of HZRM solutions with optimal solutions

For comparison purposes, we have selected three common structures for the topology inside a cloud: i) a star topology, ii) a loop (or a cycle) topology, and iii) a linear topology. The optimal probe subset for each example is found by complete enumeration of all possible solutions, since solutions of INLP formulations (found by applying standard INLP solution algorithms) are not guaranteed to be globally optimum. To be able to enumerate all possible solution alternatives, we have selected the size of the sample topologies relatively small (for each example, number of nodes is selected between 5 and 6, and number of arcs is selected to be 5). Table 3 summarizes the optimal and the HZRM solutions for the test topologies.

Table 3: Solutions of the heuristic of ZRM versus optimal solutions for the three common network structures

Topology	N	M *(unidirectional)*	# all probes	HZRM solution (# probes)	Optimal solution (# probes)	HZRM Sol. - Opt. Sol.
	6	5 (10)	30	12	10	2
	5	5 (10)	20	10	10	0
	6	5 (10)	30	12	12	0

As seen in Table 3, HZRM performed satisfactory in these common network structures by selecting optimal probe subset size in two out of three samples. In addition, HZRM achieved a solution, which is relatively close to the optimal solution in the star topology example.

4.5. Application of HZRM on vBNS topology

To demonstrate the performance of HZRM, we consider the vBNS (very high Backbone Network Service) (Figure 7). There are 12 nodes and 17 bi-directional (34 unidirectional) arcs in the network cloud. If we want to observe all delays on all the source and destination pairs in the topology, then we have to send unidirectional probes on 132 paths. The routes between nodes were selected arbitrarily using the minimum hop path. In some cases, alternative paths (with the same number of hops) were observed between some source and destination pairs. In situations with multiple minimum hop alternatives, one of the alternatives is assigned arbitrarily as the path between the source and the destination.

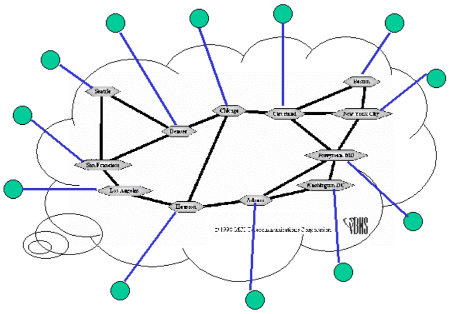


Figure 4: vBNS backbone (Source: [5])

After solving the vBNS example with HZRM, we achieved a solution subset of 32 probes. By sending probes between the 32 selected paths, we can obtain the network conditions and identify high delays on all 132 paths, thus obtaining a 75.75 percent reduction in the number of observed paths. Note that although Figure 4 indicates the topology it is not used either in the HZRM or the monitoring. In fact if we indeed assume that the topology and routing is known, then one can achieve 90.91 percent reduction in the number of observed paths [8]. The 15.16 percent additional probes account for the lack of information on routing and topology. The main strength of HZRM is its applicability in networks with no available topology and routing information. The choice between the costs of monitoring, processing and storing more paths versus the cost of obtaining the topology and routing information is dependent on the network under consideration.

5. CONCLUSION

Given the ping-like probes, we have formulated a graph-theoretic problem, namely the Zone Recovery Methodology (ZRM), to minimize the number of source-destination probes such that every zone (a possible anomaly location) in the network is uniquely identified by a pair of paths without using any information on the network topology or the routes of the packets.

In order to solve the subset selection problem using ZRM, we have presented an integer nonlinear programming formulation. Solving the integer non-linear programming formulation with standard algorithms has exponential-time worst-case complexity. Therefore, we have developed an alternative polynomial-time heuristic (called HZRM) for the graph-theoretic formulation of the ZRM.

The proposed non-linear integer programming formulation and HZRM have been compared in terms of complexity, applicability and accuracy. We have demonstrated that HZRM provides a polynomial time solution with a very satisfactory performance in terms of number of probes selected in the subset. On the three common topological structures (a star, a loop and a linear formation) HZRM achieved the optimal solutions in two out of three examples, and missing the optimal solution by two probes in the remaining example. To observe the performance of HZRM on a real topology, we have applied HZRM on the vBNS topology. We have observed a 75.75 percent reduction in the number of monitored probes to characterize the network conditions. In addition, comparing ZRM to the case which assumes that the topology and routing information is given [8], revealed that ZRM requires monitoring an additional 15.16 percent of all the paths on the vBNS topology to take into account the unknown topology and routing.

Given the final subset of probes, the network can be continuously monitored, and the delay across each arc in the network can be stored. Then the high delay between any source and destination can be detected (without monitoring the path between every source and destination pair) if any of the zones in the corresponding path has an anomaly causing high delays. Also, the hot spots can be identified by observing the anomalies in the delay across each zone.

REFERENCES

1. R.K. Ahuja, T.L. Magnanti. and J.B. Orlin, *Network flows: theory, algorithms, and applications*, Prentice Hall, 1993.
2. M.S. Bazara, J.J. Jarvis, and H. D. Sherali, *Linear programming and network flows, second edition,* John Wiley & Sons, New York, 1990.
3. R. Govindan, and A. Reddy, "An analysis of internet inter-domain topology and routing stability," *IEEE Infocom*, pp. 7b.4.1-7b.4.8., 1997.
4. http://www.advanced.org/
5. http://www.vbns.net/
6. N. Jamison, et al, "VBNS: not your father's Internet," *IEEE Spectrum*, July 1998, pp. 38-46, 1998
7. G.L. Nemhauser, and L.A. Wolsey, *Integer and comb. optimization*, John Wiley & Sons, New York, 1998.
8. H.C. Ozmutlu, R.R. Barton, N. Gautam and W. Hery, "Network Monitoring: Probe-Subset Selection Using the Constrained Coverage Problem", *Perf. and Ctrl of Network Systems Conference*, OPIE, Sept. 1999.
9. H.C. Ozmutlu, N. Gautam and R.R. Barton, "Random-hop Approach for Network Topology Generation", submitted to IEEE/ACM Transactions on Networking, 2001.
10. H.C. Ozmutlu, N. Gautam and R.R. Barton, "Managing End-to-end Network performance via Optimized Monitoring Strategies", submitted to *The Journal of Network and System Management*, Special Issue on Management of Converged Networks, 2001.
11. V. Paxson, and S. Floyd, "Wide-area traffic: the failure of Poisson modeling". IEEE/ACM Transactions on Networking 3, pp. 226-244. 1995.
12. V. Paxson, "End-to-end routing behavior in the Internet," *Proceedings of SIGCOMM '96*, pp. 25-38, 1996.
13. K. Thompson, G. J. Miller, and R. Wilder, "Wide-area Internet traffic patterns and characteristics," *IEEE Network Magazine*, November/December 1997, pp.10-23, 1997.
14. K. Varadhan, D. Estrin, and S. Floyd. "Impact of network dynamics on end-to-end protocols: case studies in TCP and reliable multicast," Submitted to *IEEE Infocom*, 1998.

15 F. Yegenoglu, F. Faris and O. Qadan. "A model for representing wide area internet packet behavior", *IPCCC'00 Proceedings of the IEEE International Performance, Computing and Communications Conference*, pp.167-175, 2000.