Network Discovery from Passive Measurements

Brian Eriksson
UW-Madison
bceriksson@wisc.edu

Paul Barford
UW-Madison
pb@cs.wisc.edu

Robert Nowak
UW-Madison
nowak@ece.wisc.edu

ABSTRACT

Understanding the Internet's structure through empirical measurements is important in the development of new topology generators, new protocols, traffic engineering, and troubleshooting, among other things. While prior studies of Internet topology have been based on active (traceroute-like) measurements, passive measurements of packet traffic offer the possibility of a greatly expanded perspective of Internet structure with much lower impact and management overhead. In this paper we describe a methodology for inferring network structure from passive measurements of IP packet traffic. We describe algorithms that enable 1) traffic sources that share network paths to be clustered accurately without relying on IP address or autonomous system information, 2) topological structure to be inferred accurately with only a small number of active measurements, 3) missing information to be recovered, which is a serious challenge in the use of passive packet measurements. We demonstrate our techniques using a series of simulated topologies and empirical data sets. Our experiments show that the clusters established by our method closely correspond to sources that actually share paths. We also show the trade-offs between selectively applied active probes and the accuracy of the inferred topology between sources. Finally, we characterize the degree to which missing information can be recovered from passive measurements, which further enhances the accuracy of the inferred topologies.

Categories and Subject Descriptors

C.2.3 [**Computer-Communication Networks**]: Network Operations—*Network Monitoring*

Keywords

Topology, Embedding, Measurement, Inference, Imputation

General Terms

Measurement, Algorithms

SIGCOMM'08, August 17–22, 2008, Seattle, Washington, USA.

1. INTRODUCTION

Discovering and characterizing Internet structure and topology through empirical measurements has been an active research area for some time (*e.g.,* [2, 15, 17, 22, 31]). These studies have helped to shed light on the huge size, intricate interconnection characteristics and complicated interplay between underlying physical topology and the traffic that flows over the infrastructure. Most prior work on measuring the Internet's structure has been based on *active measurement* techniques that use traceroute-like tools or tomographic probing, and there are several large on-going topology discovery projects based on active probe-based tools (*e.g.,* [5, 29, 21]).

There are three important limitations in the use of active probe-based tools for Internet topology discovery. First, the vast size of the Internet means that a set of measurement hosts M and target hosts N where $N \gg M$ must be established in order for the resultant measurements to capture the diverse features of the infrastructure (especially on the edges of the network [4]). Second, active probes sent from monitors to the large set of target hosts result in a significant traffic load and complex management issues. Third, service providers frequently attempt to thwart structure discovery by *e.g.,* filtering ICMP packets, which renders standard topology discovery tools like traceroute ineffective.

In this paper we investigate the problem of Internet-wide structure and topology discovery from *passive measurements*. Our objective is to develop techniques for inferring meaningful structural characteristics such as client groups and shared paths using only very simple passive measurements – specifically the source IP address and TTL fields from IP packet headers. We argue that these simple measurements can be widely collected without significant management overhead, and offer an opportunity to greatly expand the perspective of Internet structure due to the diversity of traffic observed in passive monitors [9, 10].

There are significant challenges in using passive packet measurements for discovering Internet structure. First, and most importantly, the individual measurements themselves would seem to convey almost no information about network structure. Second, source IP addresses are often considered sensitive and are typically subject to privacy constraints. We address the latter, to an extent, by only using source IP addresses as unique identifiers of hosts (*i.e.,* source IP addresses could be anonymized, as long as anonymization is consistent across measurements and monitors). Unfortunately, this further complicates the structure discovery problem. Despite these severe limitations, we demonstrate two

surprising capabilities in this paper:

1. Internet sources[1] can be automatically and accurately clustered into meaningful groups corresponding to *shared* network topology;

2. Network topology can be accurately recovered from large volumes of passive data when coupled with a very small number of additional active measurements.

Our methodology for inferring network structure from passive measurements begins by using a standard technique to determine the hop-count distance between sending hosts and passive monitors [18]. It is not uncommon to observe packets from an individual source in several of the passive monitors, resulting in a hop-count distance vector for that source. These vectors provide an indication of the topological location of the source relative to the monitors. Considering all such sources and hop-count vectors places a large number of constraints on the underlying topology relating sources and monitors.

Sources can then be clustered by examining similarities in hop-count vectors. Sources within a client group [20] or (stub) autonomous systems will have similar hop-count vectors. Thus, two sources with identical hop-count vectors are likely to be topologically close together. We also observe that sources from a common topological location may have hop-count vectors that differ only by a constant offset, owing to the fact that they may have different paths to network egress points, but then share routes to passive monitors. These offsets can be eliminated by removing the average value of each hop-count vector, resulting in what we call a *hop-count contrast*. The hop-count contrasts of two sources from the same area of the Internet should be nearly identical. Slight variations will, of course, persist due to finer scale routing variations. The resulting hop-count contrasts will therefore tend to be clustered about nominal values associated with local areas of the Internet. We use a set of simulated topologies [24] to show that clustering methods applied to the hop-counts reliably reveal such structure.

Next, we develop a lightweight method for discovering the network topology connecting sources and monitors by augmenting the passively collected data with a very small number of active measurements. Roughly speaking, the clustering process described above enables topology discovery from a number of active measurements proportional to the number of discovered clusters (*i.e.,* we need only make O(1) traceroute measurements from each cluster to each passive monitor site). Since the number of clusters is expected to be drastically smaller than the number of sources, the burden of active measurements is almost inconsequential. The active measurements provide ground-truth assessments of the number of shared hops between pairs of sources and a passive monitor or pairs of monitors and a source. This knowledge of shared hops, coupled with the clustering inferred from the passive data, suffices to reconstruct the logical network topology. We use simulated topologies and Skitter data [5] to show the trade-offs between active probe budget and accuracy for our approach.

[1]In this paper, we equate source IP addresses with individual hosts (which we refer to as "sources"), understanding that this could introduce some error in the accuracy of topology estimation.

Source clustering and topology discovery both depend on the quality of the hop-count data. Due to the passive nature of the data collection process, typically packets from a source will only be observed at a (small) subset of the passive monitors. The resulting hop-count vector will be incomplete, with missing entries corresponding to the monitors that did not observe packets from the source. The missing data greatly confounds the clustering process and subsequent topology discovery. To cope with this serious issue, we adopt a probabilistic model for the hop-count contrasts. Since we expect the contrasts to cluster, a mixture of Gaussian densities is used to approximate the distribution of contrasts. Each component of the mixture is intended to represent one of the clusters. The parameters of the mixture density can be fitted to the (incomplete) hop-count data using a clever iterative procedure due to Ghahramani and Jordan [13]. Moreover, the resulting mixture density then provides a principled mechanism for imputing the missing data and accurately clustering sources. We use simulated and empirical data to show the relationship between the accuracy of our method and the quantity of missing data.

The remainder of this paper is organized as follows. In Section 2, we review prior work related to our study. In Section 3 we describe the data sets used in our experiments. In Section 4, we describe our source clustering algorithm. In Section 5, we show how a very modest number of active measurements provides enough additional information to recover the topology relating sources and passive monitors. In Section 6 we tackle the issue of missing data and demonstrate that accurate clustering is still possible even when the passive data are highly incomplete. We conclude and describe future work in Section 7.

2. RELATED WORK

Internet structure can be considered in a number of ways including connectivity (*e.g.,* between autonomous systems, between IP addresses, between routers or between POP's [32]), distance related properties (*e.g.,* geography [16, 19], packet latency [11, 12]), or behavioral characteristics (*e.g.,* social network membership). The focus of our work is on identifying Internet structure in terms of clusters of clients [20] and shared paths [6] toward the goal of full router-level connectivity identification [15]. Our work differs from prior studies of client clusters in that we do not rely on IP address details. Prior studies of shared paths and router topologies have used active probe-based measurements exclusively while our work is focused on using primarily passive measurements. While passive measurements of routing updates can be used to establish intra-domain network maps [26], our goal is to discovery Internet-wide structure with much more simple measures.

A related perspective is afforded by coordinate systems, which have been proposed as a means for estimating latency between arbitrary hosts in the Internet [25, 33, 7]. Coordinate systems rely on latency measurements between a set of landmark nodes to create an embedding in a high dimensional space. Hosts can then use estimates of their latency to points in the coordinate space to predict the latency to hosts in the Internet. The challenges in creating coordinate systems are in making them scalable, robust and accurate. One of our topology discovery techniques is based on the idea of establishing a topology framework via active measurements, which is similar to landmarks. Another study that bears

some similarity to ours is by Shavitt and Tankel who develop the idea of a hyperbolic embedding which includes the idea of Internet structure in distance estimation [30].

Passive measurements of packet traffic can be gathered by deploying specialized hardware on TAP'ed links (*e.g.,* [9, 10]). While measurements from TAP'ed links could be used in our work, publicly available data sets almost always anonymize source IP addresses making it impossible to relate measurements from multiple sites. An alternative form of passive packet measurements are those collected in network honeypots [1, 3, 28, 34]). Honeypots monitor routed but otherwise unused address space, so all traffic directed to these monitors is unwanted and almost always malicious. Honeypots do not solicit traffic, however low interaction sensors will respond to incoming connection requests in order to distinguish spoofed addresses. In this way they are not completely passive. However, monitors of large address segments can receive millions of connections per day from systems all over the world and therefore offer an incredibly unique and valuable perspective [27]. The unsolicited nature of honeynet traffic coupled with the volume and wide deployment of monitors make it an attractive source of data for our work.

Finally, we proposed the idea of using passive measurements as the basis for network discovery and present initial results on imputing missing data in an extended abstract in [8]. We expand and generalize that work by developing an algorithm for client clustering, by developing methods to infer topology and shared paths that use a small number of active probes, and evaluate our algorithms with simulated and empirically derived maps of the Internet.

3. PASSIVE HOP-COUNT DISTANCE MEASUREMENTS

We use three different data sets to evaluate the algorithms that are described in this paper. The first are a set of topologies generated by Orbis [24]. Orbis is one of the latest and most realistic network topology generators. It creates graphs that have properties that are consistent with many of those observed in the Internet. The Orbis-generated synthetic networks enable us to analyze the capabilities of methods with full ground truth and over a range of sizes.

The second data set that we use in this paper is an router-level connectivity map of the Internet based on data collected by Skitter [5]. Measurements in Skitter are based on traceroute-like active probes sent from a set of 24 monitors to a set of nearly 1M target hosts distributed throughout the Internet. We use the openly available router-level map create from data collected between April 21 and May 8, 2003. This map consists of 192,224 unique nodes and 609,066 undirected links. It is important to note that the goal of the Skitter target host list is to have one responding node in each /24 prefix. Thus, the characteristics of the Skitter graph with respect to destination subnets is different from Orbis generated topologies, which reflect collections of nodes in subnets.

The third data set used in our study was collected over a 24 hour period starting at 00:00 on December 22, 2006 from 15 topologically diverse honeypot sensors. These sensors are located in 11 distinct /8 prefixes that are managed by 10 different organizations. The segments of IP address space monitored by the honeypots varied from /25 to /21 plus one /16. Over 37,000,000 total packets were collected and evaluated in our study. The packets do not contain spoofed source IP addresses since they were the responses to SYN/ACKs from the honeynet [3]. Details of the data set can be found in Table 1. In order to preserve the integrity of the honeypots, we cannot disclose their locations in IPv4 address space.

Table 2: Counts of occurrences of common source IP addresses in multiple honeypots

Num. Honeypots	Num. Sources
2	8680
3	4051
4	2816
5	2156
6	1570
7	1583
8	1574
9	55
10	4

Of particular interest and importance in our evaluation are the occurrences of the same source IP address in multiple honeypots. We found that 93.5% of the unique IP addresses in our data set appear in only one of the honeypots. This is most likely due to the diverse locations of the sensors coupled with the fact that different instances of malware limit their scans to smaller segments of address space. Nevertheless, this left us with over 22,000 unique IP addresses from which we conducted our analysis. Details of the instances of multiple occurrences of unique IP addresses are listed in Table 2 (note that there were virtually no addresses were seen in more than 10 monitors).

Our analysis assumes that the only data that will be used to infer network structure is the source IP address (used only to uniquely identify a host and as an active probe target) and TTL extracted from the header of each packet. In the case of the Orbis and Skitter data sets, we synthesize these values. In the case of the honeynet data we use the clever technique described in [18] to infer the number of hops between the honeypot monitor and the host. This inference is made based on the fact that (*i*) there are only a few initial TTL values used in popular operating systems (*e.g.,* 64 for most UNIX variants, 128 for most Microsoft variants and 255 for several others), and (*ii*) typical hop counts for end-to-end paths are far less than the differences between the standard TTL values. Thus, hop count is inferred by rounding the TTL up to the next highest initial TTL value and then subtracting the initial TTL.

3.1 Passive Measurement Infrastructure

We assumed that the ground truth router-level topology of the Internet will resemble the network in Figure 1-(left) [4]. In this diagram, packets sent from sources S_i will depart from the edge of the network and eventually enter the densely-connected core component through a border router. The packets will traverse the core, exit through another border router and eventually be intercepted by a passive monitor M_j. This configuration enables edge and core mapping, and assumes monitors such as honeynets or passive collection near *e.g.,* busy web servers. An alternative configuration could include passive monitors in the core rather than

Table 1: Details of honeypot data sets used in our study. All data was collected over a one day period on December 22, 2006.

Node	Total Pkts.	Uniq. IPs	Mean Hops	Hop Std. Dev.
1	22,586,386	217,505	13.16	9.35
2	10,533,700	9,554	7.95	7.34
3	100,689	8,431	12.11	7.72
4	4,446	738	11.45	15.34
5	25,062	474	12.07	7.32
6	128,158	6,423	6.87	7.67
7	110,621	11,942	17.23	9.00
8	49,253	6,456	15.49	10.26
9	42,226	6,534	14.31	10.72
10	45,334	6,223	13.93	9.73
11	75,522	8,645	12.45	11.14
12	523,907	8,714	9.82	9.82
13	1,955,100	6,195	19.24	8.46
14	332,986	5,364	8.16	8.68
15	917,894	107,632	12.79	6.42

only the edge of the network.

Assuming this structure of the network, one can partition the total layer 3 hop count distance values into the distance (or number of router hops) from the IP source S_i to the first core border router b, and the distance from b to the measurement node M_j. We define the variables $\{x_i =$ the number of layer 3 hops along the path from source S_i and the first core border router $b\}$, and $\{w_{i,j} =$ the number of layer 3 hops between the first core border router b of source S_i and measurement node $M_j\}$. This allows us to partition the hop count distance values into the two separate paths, $h_{i,j} = x_i + w_{i,j} =$ the number of layer 3 hops between source S_i and measurement node M_j.

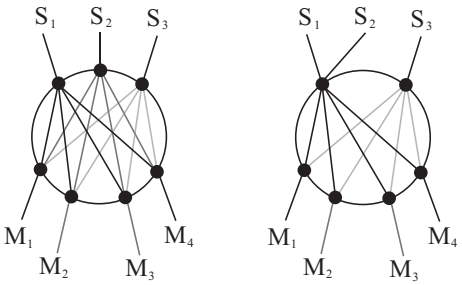

Figure 1: (Left) Example Network Topology with sources S_i sending packets through a core component to monitors M_j, (Right) Example network where S_1 and S_2 share a border router.

Now consider the situation where two sources (S_i, S_j) are connected at the same border router (see Figure 1-(right)). Given that these two IP sources will share a path through the core to each measurement node, we can state:

Theorem 1. Given two sources (S_i, S_j) sharing a common core ingress border router, then $h_{i,k} - h_{j,k} = C$ for all measurement nodes M_k with paths through the core (for some integer constant C).

PROOF. Given hop count distance values $h_{i,k} = x_i + w_{i,k}$ and $h_{j,k} = x_j + w_{j,k}$. For any measurement node M_k, with both S_i and S_j having paths through the core to the measurement node, there will be a common path for both IP sources from the border router to the measurement, such that $w_{i,k} = w_{j,k}$. Therefore, $h_{i,k} - h_{j,k} = x_i - x_j + (w_{i,k} - w_{j,k}) = x_i - x_j = C : \forall k$. □

3.2 Hop Count Distance Vectors and Network Topology

In [8], it was shown that hop count distance vectors that are similar/close in a Euclidean sense, do not necessarily translate to IP sources that are close in the actual network topology. Thus, the exclusive use of raw hop count distance vectors for clustering could place IP sources that are actually far apart in the same group. One reasons for this is that clustering the raw hop count distance data ignores the network-centric knowledge embedded in the distance vectors. To exploit the integer distance offset property of the clusters of IP sources that shared border routers, we perform preprocessing on the hop count distance vectors such that if h_i and h_j share a common border router, then after some transformation, the two vectors are equivalent. The preprocessing here takes the form of converting the hop count *distance vectors* ($\mathbf{h_i} = [\mathbf{h_{i,1}}, \mathbf{h_{i,2}}, ..., \mathbf{h_{i,M}}]$) to hop count *contrast vectors* ($\mathbf{h'_i}$), where the mean value of each vector is subtracted from each element of the hop count distance vector.

$$\mathbf{h'_i} = \mathbf{h_i} - \boldsymbol{\mu_i}\mathbf{1}$$

Where $\boldsymbol{\mu}_i = \frac{1}{M}\sum_{k=1}^{M} h_{i,k}$ and $\mathbf{1} = [\mathbf{1}, \mathbf{1}, ..., \mathbf{1}]$. Using Theorem 1, we can state with certainty that if $h_{i,k} - h_{j,k} = C : \forall k$, then $h'_{i,k} = h'_{j,k} : \forall k$.

4. CLUSTERING IP SOURCES

The first goal of our work is to develop a method for generating clusters of IP sources that are topologically close to each other from a layer 3 hop count perspective. The observation that is key to our algorithms is that the location of a given source S_i is defined by it's relative distance to multiple monitors $M_j \ldots M_k$, and that sources with similar relative distances will be topologically close to each other (assuming

that packets from the sources are observed in a sufficient number of monitors). In this section we describe our clustering methodologies and demonstrate their capability using synthetically generated network maps.

4.1 Client Clustering

We can generate clusters of IP sources using unique hop count contrast vectors and the simple K-Means algorithm. Experiments with synthetic topologies showed that clusters of various sizes could be generated (K-Means requires that the number of clusters be specified a priori) with a clear trade off between the number of clusters and the number of sources included in each cluster. A larger number of small clusters with minimal differences between contrast vectors might be considered a "good" choice with this approach.

Unfortunately, these small clusters miss the case where sources located in the same area (which we will refer to as a "subnet" although this is not related to IP address structure) of the network have differing hop count contrast vectors. This situation occurs when one or more monitor nodes are located in the same subnet as the cluster, or when the subnet has multiple egress points. This sort of subnet topology produces variability in the contrast vectors. This observation suggests that rather than clustering sources according to unique contrast vectors, clusters that allow for a bit of variation about a nominal value may better capture subnets of sources.

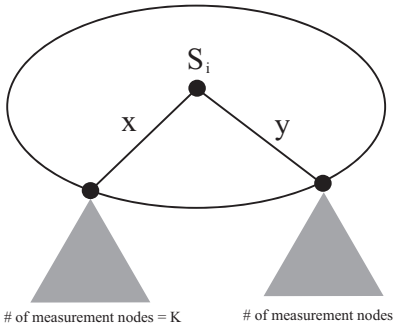

Figure 2: Example of a subnet having multiple egress points.

Consider the subnet topology in Figure 2, where there are two egress points to the set of measurement nodes for each IP source located in the subnet. The first egress router will send paths from subnet sources to k measurement nodes M_1 to M_k, and the second egress router will route paths from subnet sources to j measurement nodes M_{k+1} to M_M (where $M = j + k$). Every source will have a (potentially unique) path of length x to the first egress router, and (potentially unique) path of length y to the second egress router. For the paths from the egress router to the measurement nodes, the paths will be common for all sources in the subnet. Using this setup, we can state Theorem 2.

Theorem 2. Given a subnet with two egress points (as in Figure 2), all sources contained in the subnet will have collinear hop count contrast vectors.

PROOF. We first define the nominal distance vector ($\mathbf{h}$) as the distances from the egress routers to the measurement nodes, where $\mathbf{h_1}$ is the k-length vector containing the distances from the border node to the first k measurement nodes, and $\mathbf{h_2}$ is the j-length vector with the distances from the border node to the last j measurement nodes.

$$\mathbf{h} = \left[\begin{array}{cc} \mathbf{h_1} & \mathbf{h_2} \end{array} \right]$$

We can state the hop count distance vector for each source in the subnet as the addition of the intra-subnet paths x, y and the nominal distance vector, where $\mathbf{1_k}$ is the k-length all ones vector:

$$\mathbf{h_i} = \left[\begin{array}{cc} x \cdot \mathbf{1_k} & y \cdot \mathbf{1_j} \end{array} \right] + \left[\begin{array}{cc} \mathbf{h_1} & \mathbf{h_2} \end{array} \right]$$

We define the nominal contrast vector as:

$$\mathbf{h'} = \mathbf{h} - \boldsymbol{\mu}_\mathbf{h} = \left[\begin{array}{cc} \mathbf{h_1} & \mathbf{h_2} \end{array} \right] - \boldsymbol{\mu}_\mathbf{h}$$

Therefore, each IP source located in the subnet will have contrast vector:

$$\begin{aligned} \mathbf{h'_i} &= \mathbf{h'} + \left[\begin{array}{cc} x \cdot \mathbf{1_k} & y \cdot \mathbf{1_j} \end{array} \right] - \left(\frac{k}{M} x + \frac{j}{M} y \right) \mathbf{1} \\ \mathbf{h'_i} &= \mathbf{h'} + \left[\begin{array}{cc} \frac{j}{M}(x-y)\,\mathbf{1_k} & \frac{k}{M}(y-x)\,\mathbf{1_j} \end{array} \right] \end{aligned}$$

Setting $r = x - y$, the difference between the IP source hop contrast vector and the nominal contrast vector is:

$$\mathbf{h'_i} - \mathbf{h'} = \left[\begin{array}{cc} \frac{j \cdot r}{M}\mathbf{1_k} & -\frac{k \cdot r}{M}\mathbf{1_j} \end{array} \right]$$

Therefore, all IP sources sharing the same egress routers will have collinear contrast vectors. □

From Theorem 2, we see that sources in subnets with multiple egress points may have slight variations in the hop count vectors. The precise nature of these variations depends on several uncertain factors, including the number of egress points and the nature of the paths to the egress points. Thus, we will account for this uncertainty with a probabilistic model for the variability in hop count contrasts of sources within a subnet.

4.2 Gaussians Mixture Model for Subnet Clusters

While the exact nature of the distribution hop count contrast vectors for sources in a given subnet is unknown, a multivariate Gaussian model is perhaps the simplest way to capture the variability of the data. The covariance matrix can account for structure in the distribution, such as the collinearity discussed in Theorem 2, as well as other correlations arising the idiosyncracies of routing internal to the subnet. Since the hop count data includes sources from many different subnets, the overall distribution of hop count contrast vectors can be modeled with a mixture of Gaussian models, in which each Gaussian component represents the distribution within one subnet. An example of these Gaussian clusters can be seen in Figure 3, where a two dimensional histogram of hop count contrast vectors are shown with possible clusters shown by the drawn ellipses.

Gaussian mixture models can be fitted to data using the well known Expectation-Maximization (EM) algorithm, and in particular the version proposed in [23] automatically determines the proper number of clusters using an information-theoretic criterion. Once the Gaussian mixture model is determined, each hop count contrast vector will be associated most significantly with a given Gaussian component. This then provides a clustering of the sources, where the number clusters is equal to the number of Gaussian components inferred by the EM algorithm [23]. Moreover, we will see later (in Section 6.1.2) that the Gaussian mixture model and EM

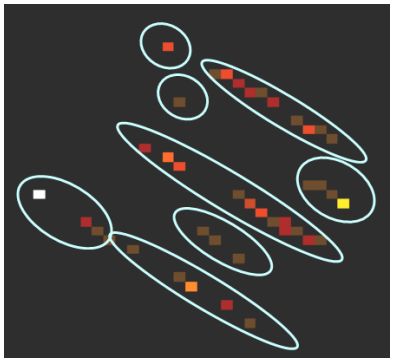

Figure 3: 2-D histogram of hop count contrast vectors with clusters highlighted in ellipses.

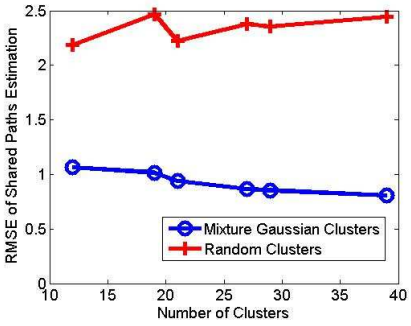

Figure 4: Comparison of Gaussian mixture clusters to random clusters. Simulated topology, $N = 1000$, $M = 8$.

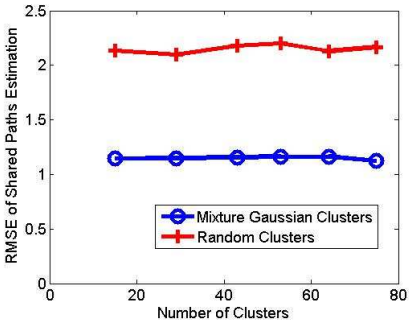

Figure 5: Comparison of Gaussian mixture clusters to random clusters. Skitter topology, $N = 700$, $M = 8$.

algorithm provide a powerful tool for imputing missing hop count data.

4.2.1 *Subnet Cluster Analysis*

To assess the topological relevance of the clusters determined by the Gaussian mixture model, we consider the problem of shared infrastructure estimation (to be discussed in detail in Section 5.2). The topology relating the sources in a given cluster to the measurement nodes can be estimated by selecting one source from the cluster and performing traceroute measurements from this source to each measurement node. If the all the sources in the cluster share the same paths, then this estimate is perfect. We do not, however, expect this to be the case, even for sources located in the same subnet, for the reasons state above. Nonetheless, these routes should provide good predictions for the routes, if the clusters are topologically meaningful. The accuracy of the predictions is measured by calculating the error in predicted shared hops in the paths between pairs of sources and a measurement node (we define Root Mean Squared Error in Section 5). The error rates in the predictions of shared path lengths are shown in Figures 4 and 5, comparing the performance of the predictions based on the Gaussian mixture clusters with that of predictions based on randomly clustered sets of sources. The clusters determined by the Gaussian mixture model result in significantly better predictions, indicative of the fact that they are indeed grouping sources that have share similar paths to the the measurement nodes.

5. TOPOLOGY DISCOVERY

The source clusters identified by our algorithm are topologically meaningful. However, they do not reveal the topological relationship between the shared paths from the clusters to the measurement nodes. In this section, we will show that by coupling the passive hop count data with a small number of active measurements, we can identify the topological relationships between clusters. The active measurements will take the form of traceroutes from the measurement nodes to a *small subset* of target hosts which effectively act as representatives for the clusters. This is in contrast to the *e.g.,* the Skitter methodology [5], where active measurements are taken from all measurement nodes to a large set of target hosts.

5.1 Cluster-Level Topology Discovery

Given a set of IP source clusters, discovering shared topology between clusters becomes a straightforward task. For every cluster, randomly choose an IP source in the cluster and perform active traceroute measurements between that IP source (consider as a representative for its cluster) and the set of measurement nodes. If the clusters are topologically significant, the topology will have been discovered.

There are at least two potential problems with this straightforward approach to topology discovery. First, the source clusters may not be completely correct from a topological perspective, due the possible existence of multiple egress points and missing hop counts (the missing data issue will be thoroughly addressed in Section 6). Second, from an Internet-wide perspective, the number of clusters may still be prohibitively large for exhaustive (cluster-wise) traceroute probing.

5.2 Shared Infrastructure Estimation

Given the drawbacks to the deterministic cluster-to-cluster topology discovery technique, we address the problem of estimating shared infrastructure between pairs of IP sources.

5.2.1 *The Canonical Subproblem*

Consider a *triple* $\{S_i, S_j, M_k\}$, where two sources have a path to a single measurement node as seen in Figure 6.

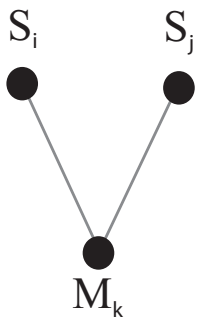

Figure 6: The canonical subproblem: two IP sources connecting to a single measurement node

There are three possible potential topologies connecting this triple (two sources to one measurement node), with a sharedness spectrum ranging from absolutely no sharedness with two separate paths from each source to the measurement node, to complete sharedness with both sources on a single path to the measurement node, with the intermediate stage of some length of shared path between the two sources. It is easy to verify that if the number of shared hops is known for all such canonical subproblems, then the logical topology relating the sources to the measurement nodes can be determined. This follows by observing that the set of paths from the sources to a given measurement node form a tree. Therefore, this section will focus on estimating $P(i,j,k)$, the length of the shared path between two IP sources i,j to a single measurement node k using the passive data and a limited number of traceroute measurements.

5.2.2 Cluster-Level Shared Path Length Estimation

Toward the goal of cluster-level topology discovery, one can discover shared path lengths by using active measurements from a single representative of each cluster as seen in Figure 7. We assume that all sources in C_i will share the same path of length x to measurement node M_k with all sources in C_j. Therefore, for a single active measurement of each cluster, we have an estimate of the shared path lengths between IP sources contained in all other clusters in the topology.

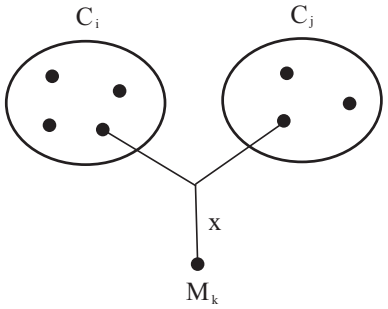

Figure 7: Example of cluster-level path estimation.

5.2.3 Predictive Shared Path Length Estimation

For two hop count distance vectors, it is necessary to first develop some metric for the amount of sharedness between the two vectors. The similarity of the hop count contrast vectors indicates the likelihood that the two sources are within the same subnet. The greater the similarity the stronger the evidence for shared infrastructure in the paths to the measurement nodes. To assess the potential for shared infrastructure to a given measurement node we consider the difference in hop count distances to that node and calculate the number of other measurement nodes that result in the same hop count difference. Formally, we define,

$$U_{i,j,k} = |\mathbf{T_{i,j,k}}| \tag{1}$$

Where $\mathbf{T_{i,j,k}} = \{k' : |h_{i,k'} - h_{j,k'}| - |h_{i,k} - h_{j,k}| < \epsilon\}$ (for $\epsilon > 0$)

As the value of $U_{i,j,k}$ becomes closer to the number of measurement nodes, there is a higher likelihood of a longer shared path to each measurement node.

Given the training set $\mathbf{I_k}$, where each element is the index of an IP source for which we have exact knowledge (from active measurements) of the labeled path to measurement node M_k, we can then construct sets consisting of pairs of training nodes that share the same offset value for the particular measurement node k.

$$\mathbf{I_k^c} = \{[x,y] : x,y \in \mathbf{I_k}, U_{x,y,k} = c\} \tag{2}$$

Considering two paths from S_i to M_k and S_j to M_k, we can state that the shortest shared path would be of length zero as shown in Figure 8-(left), and the longest shared path would be of length $= \min(h_{i,k}, h_{j,k})$ as shown in Figure 8-(right).

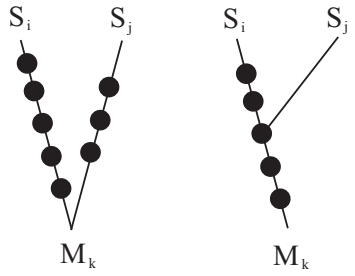

Figure 8: Potential topologies for two sources to one measurement node (each black dot represents a router hop) (Left) - Shortest possible shared path, (Right) - Longest possible shared path

Given this range of shared path lengths, we can estimate the shared path length for any two sources i,j to any measurement node k by attenuating the longest possible shared path length $(= \min(h_{i,k}, h_{j,k}))$ by some value less than one, represented by α.

$$\widehat{P}(i,j,k) = \alpha \cdot \min(h_{i,k}, h_{j,k}) \tag{3}$$

The problem becomes estimating the value of α. Given some collection of training data where active measurements give observed values for the shared path lengths, we can estimate α as a function of the passive measurements of S_i and S_k. We hypothesize that the more hop count distance values that are a constant integer apart, the more sharedness that will be observed along the path. This results in learning a function whose domain is the number of hop count elements where the two vector h_i and h_j are a constant integer apart.

We can then learn the attenuation function by taking the average of the observed path lengths for each integer offset value. Therefore, for each measurement node k and uniformity metric value c:

$$\alpha(c,k) = \frac{1}{|\mathbf{I}_\mathbf{k}^\mathbf{c}|^2} \sum_{i \in \mathbf{I}_\mathbf{k}^\mathbf{c}} \sum_{j \in \mathbf{I}_\mathbf{k}^\mathbf{c}} \frac{P\left(\mathbf{I}_\mathbf{k}^\mathbf{c}(i), \mathbf{I}_\mathbf{k}^\mathbf{c}(j), k\right)}{\min\left(h_{\mathbf{I}_\mathbf{k}^\mathbf{c}(i),k}, h_{\mathbf{I}_\mathbf{k}^\mathbf{c}(j),k}\right)} \quad (4)$$

Finally, we combine the learning attenuation function to create an estimator of the shared path length for each pair of IP sources,

$$\widehat{P}(i,j,k) = \alpha(U_{i,j,k}, k) \min(h_{i,k}, h_{j,k}) \quad (5)$$

5.3 Shared Path Estimation Analysis

In Table 3, we show the results for three different methods for shared path length estimation. The methods include :

1. *Unique Contrast cluster-level Estimation* - Cluster-level estimation is performed on clusters where each represents a unique hop count contrast vector in the passive dataset.

2. *Cluster-level estimation using Gaussian mixture model* - Cluster-level estimation is performed on clusters found using the Mixture Gaussian algorithm.

3. *Predictive Function Estimation* - Using Equation 5, the estimated shared path lengths are found.

The results are based on a 1000 node synthetic topology, which was generated by Orbis [24]. We randomly select 800 leaf nodes (sources) and 8 measurement nodes in the graph, and assume "complete data" *i.e.,* that probes from all sources are received at all measurement nodes. The error metric that we use to assess the estimation accuracy the Root Mean Squared Error (RMSE) is defined as:

$$RMSE(\hat{h}) = \sqrt{\sum_{i,j} |h_{i,j} - \hat{h}_{i,j}|^2} \quad (6)$$

Where an RMSE of x indicates that the estimated shared number of hops is on average x hops away from the true number of hops extracted from the graph.

The results show that the estimation from the unique contrast clustering performed the best, but required a larger number of active measurements. Using the information-theoretic approach from [23], 7 clusters were found, (in comparison to 47 active measurement needed for each measurement node if performing unique contrast clustering). Using the Gaussian mixture clustering, the predictive method outperforms the cluster-level method. Simulations with different synthetic topologies provided similar results.

Estimation Type	# Clusters	RMSE
Cluster-level	7	1.28
Predictive Function	7	1.00
Unique Contrast	47	0.70

Table 3: Shared path estimation results for a 1000 node synthetic topology assuming that probes from 800 randomly selected source nodes were observed in 8 randomly selected monitors.

In Figure 9, we assess how increasing the number of clusters affects the performance of the Gaussian mixture EM cluster-level algorithm from a RMSE perspective. For this simulated synthetic topology (with $N = 800$, $M = 24$, in contrast to $M = 8$ results in Figure 4), the addition of more clusters (and hence more active measurements needed) causes a significant decrease in the error rate of the path length estimation.

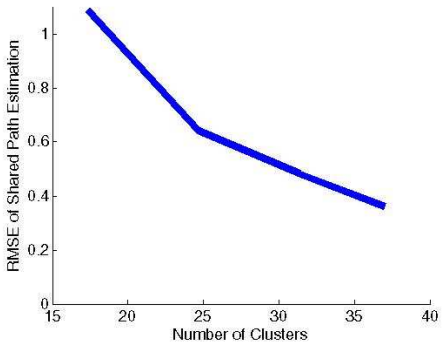

Figure 9: The effect of increasing the number of clusters on the shared path estimation performance on the simulated topology using the cluster-level shared path estimation method.

In Table 4, we show how the same three methods for shared path length estimation considered above perform on the Skitter topology described in Section 3. A random set of 700 leaf nodes were selected as sources and 8 leaf nodes were randomly selected measurement nodes.

Similar to the simulated topology, the estimation from the unique contrast clustering performed the best from an RMSE perspective but also required the largest number of active measurements. Using the information-theoretic approach, 9 clusters were found, requiring 18 active measurements of the topology for each measurement node. Again, the predictive function outperforms the cluster-level method when considering the smaller number of clusters found by the Gaussian mixture EM algorithm.

Estimation Type	# Clusters	RMSE
Cluster-level	9	1.25
Predictive Function	9	1.23
Unique Contrast	434	0.66

Table 4: Shared path estimation results for the Skitter topology assuming that probes from 700 randomly selected source nodes were observed in 8 randomly selected monitors.

In Figure 10, we assess how increasing the number of clusters affects the performance of the Gaussian mixture EM cluster-level algorithm from a RMSE perspective. For the Skitter topology (with $N = 700$, $M = 24$ (in contrast to the results for $M = 8$ in Figure 5), the addition of more clusters causes a decrease in the error rate of path length estimation, but not as significant a decrease as seen in the simulated topology example.

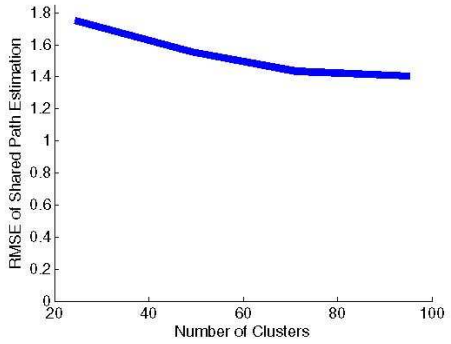

Figure 10: The effect of increasing the number of clusters on the shared path estimation performance on the Skitter topology using the Gaussian mixture EM cluster-level shared path estimation method.

6. THE MISSING DATA PROBLEM

In Section 5, we evaluated our methods for estimating shared infrastructure between IP sources with hop count distance values between all sources and measurement nodes. However, it is unlikely that packets from large numbers of IP sources will be seen at a random set of widely measurement nodes. Thus for a given set of measurement nodes (such as the honeypots describe in Section 3), there will be some number of hop count distance observations missing from the observed set. For each IP source i, we have a (potentially incomplete) hop count vector $h_{i,\mathbf{I}^{(\mathbf{i})}_{\mathbf{known}}}$ where $\mathbf{I}^{(\mathbf{i})}_{\mathbf{known}}$ is a known subset of the indices of the complete set of measurement nodes for IP source i.

We assume that this data is Missing-at-Random (where the missing data locations are chosen at random). With some assumptions on the topology and statistical techniques, we have developed a method for imputing this missing data.

6.1 Imputation Methods

6.1.1 Network-centric Imputation

Given missing data, one can easily conceive of simple imputation methods, such as imputing based on the mean value of the element, or using the nearest neighbors based on the observed elements of the distance vector. One problem with such a simplistic approach is that it does not take advantage of the structural characteristics of the network. As describe in [8], we can exploit the observed hop count distances from sources that are an integer offset from a source missing a hop count distance value. This method can be considered analogous to using nearest neighbor imputation on the hop count contrast vectors. For N sources and M measurement nodes in the network, this imputation method has computation complexity $O\left(N^2M\right)$. In the following, we describe a new method for imputing missing data that improves on the prior, network-centric approach in instances when larger fractions of data are missing.

6.1.2 Gaussian Mixture EM Imputation

In Section 4.2, we reasoned that a mixture of Gaussians model encapsulates the variability found in the hop count contrast vectors. In [14], a Gaussian mixture EM algorithm was purposed to both learn the parameters (mean, variance, prior probabilities, responsibilities) for a group of Gaussian distributions given a set of incomplete data, and then use those estimated Gaussian mixtures to impute the missing data values. The only necessary parameter input to this algorithm is the number of Gaussian mixtures to use. From [23], an information-theoretic technique was purposed to determine, given a set of complete data, how many Gaussian mixtures to use to model the data. This method is a hybrid two-step iterative approach, where the first step consists of estimating the number of Gaussians from the imputed data using the method from [23], and the second step then estimates the new imputed data values using the method from [14]. For N sources and M measurement nodes in the network, this method has computation complexity $O\left(iKNM^4\right)$, for i iterations and K Gaussian modes.

6.1.3 Imputation Performance Analysis

Using the honeynet dataset described in Section 3, we can synthetically generate missing data examples by considering the sources that are seen in M measurement nodes and knocking out (eliminating) a random subset of the hop count measurements for each hop count vector. Where X observed measurement nodes refers to each hop count vector observing X randomly selected hop counts with the rest of the vector incomplete. The new imputation method (Gaussian mixture EM) results are compared in terms of Root Mean Squared Error (RMSE, Equation 6) in Figure 11 against a naive mean method. In this analysis, we consider measurements from sources that were observed in 16 honeypots for the synthetic topology with three different sizes ($N = 1000, 2000, 3000$). The results shows a clear advantage to using the Gaussian mixture imputation method for even a small amount of observed measurements.

6.2 Shared Path Estimation with Imputed Data

Next, we assess how topology estimation is affected by using imputed data. Following the method from Section 5.2.2, the incomplete case derives clusters from imputed hop contrast vectors (using the Gaussian mixture imputation algorithm) and then performs active measurements on the clusters. The use of imputed hop contrast vectors introduces more uncertainty on the topological significance of the clusters.

To derive the estimated shared path length estimation for using incomplete data, we follow the derivation from Section 5.2.3, replacing all occurrences of the complete hop count distance vectors h_i, with the imputed hop count distance vectors $\widehat{h}_i$.

$$\widehat{P}\left(i,j,k\right) = \alpha\left(\widehat{U}_{i,j,k},k\right)\min\left(\widehat{h}_{i,k},\widehat{h}_{j,k}\right) \tag{7}$$

6.3 Topology Estimation Performance with Imputed Data

Using synthetic topologies generated by Orbis, we assess the impact of missing data imputation on topology estimation. We generate three different synthetic topologies with 1000, 2000, and 3000 nodes. The measurement nodes are randomly chosen from the set of leaf nodes in the topology, with the passive measurements simulated as the length of the shortest path found in the topology between the IP sources and the measurement nodes. After imputation of

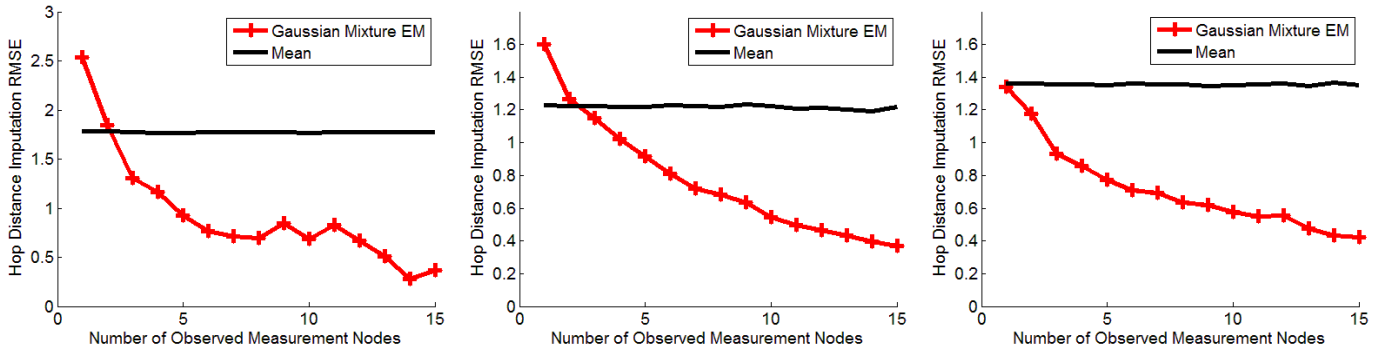

Figure 11: Imputation accuracy over a range of randomly selected missing values using data from $M = 16$ honeypots. (Left) $N = 1000$, (Center) $N = 2000$, (Right) $N = 3000$

the missing data using the mixture of Gaussians technique, 10 clusters are found in the imputed hop count contrast vectors, resulting in a active probe budget of 20 active measurements per measurement node. Figure 12 shows that the estimation of shared paths using the predictive methodology from Section 5 is comparable to the cluster-level deterministic estimation method for a majority of missing data percentages. The RMSE error rate for the estimated path lengths is defined as:

$$RMSE(\hat{P}) = \sqrt{\sum_{i,j,k} |P(i,j,k) - \hat{P}(i,j,k)|^2}$$

When the simulated topology is expanded to 2000 and 3000 total nodes, the effects on the shared path estimation algorithms can be seen in Figures 13. As seen from the figures, the increase in graph size improves upon the estimation for both algorithms.

We also consider the impact of imputation using the Skitter dataset. We selected a subset of 700 leaf nodes as IP sources and randomly selected other leaf nodes as measurement nodes. Data representing passive measurements was derived from the graph and the shared infrastructure was estimated for different levels of missing data. In Figure 14, we see the performance of the two shared path length estimation methods on the Skitter topology for $(M = 8, 16, 24)$ and 10 clusters found. In contrast to the results from the synthetic topologies, the predictive estimation method performs better than the cluster-level method. This can be explained by the fact that Skitter measurements typically only have a single leaf node for each branch of the topology. This puts the Cluster-level method at a disadvantage for estimating shared topology.

7. CONCLUSIONS AND FUTURE WORK

In this paper, we address the problem of identifying network structure and topology through the use of passive measurements of IP packet traffic. While prior work on client clustering, shared path estimation and router-level topology mapping has been based on information embedded in IP addresses or via active-probes, passive measurements offer an opportunity to expand perspective and reduce traffic and management overhead. However, the minimal information (source ID's and hop counts) considered in our problem formulation presents serious challenges to identifying meaningful network structure. We describe a method for clustering hosts based on Gaussian mixtures of hop count vectors. We demonstrate the trade-offs between cluster size and shared path length using a set of synthetic topologies. We then describe two topology estimation techniques that rely on a small set of active probe-based measurements. The first method established ground truth paths between clusters via active probes, and the second uses a predictive approach to estimate shared path lengths based on a topology framework established with active probes. The capabilities of both methods are evaluated using synthetic and empirical network maps. Finally, we describe an imputation method for estimating missing data in passive measurements, and demonstrate the capabilities of this method using data collected in a set of network honeypots distributed around the Internet.

This study represents a first step toward our goal of accurately identifying Internet topology with passive measurements. There are several important next steps that we intend to address in future work. The first is to construct larger graphs of the Internet using datasets from honeynets [1] and other sources such as web servers that receive a great deal of traffic, and to validate these extensively with data sets from Internet mapping projects such as Skitter. We also plan to develop additional methods for enriching our graphs, for example, by coupling them with other data such as BGP routing information, and to consider how dynamic changes in hop counts can be accommodated in our formulations. Finally, we plan consider problems related to measurement such as optimal monitor placement, and reduction or elimination of active measurements for shared topology estimation.

Acknowledgments

The authors would like to thank Farnam Jahanian and Michael Bailey from the IMS project at University of Michigan for providing the data used in this study. This work was supported in part by NSF grants CNS-0347252, CNS-0646256, CNS-0627102 and CCR-0355653. Any opinions, findings, conclusions, or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the NSF.

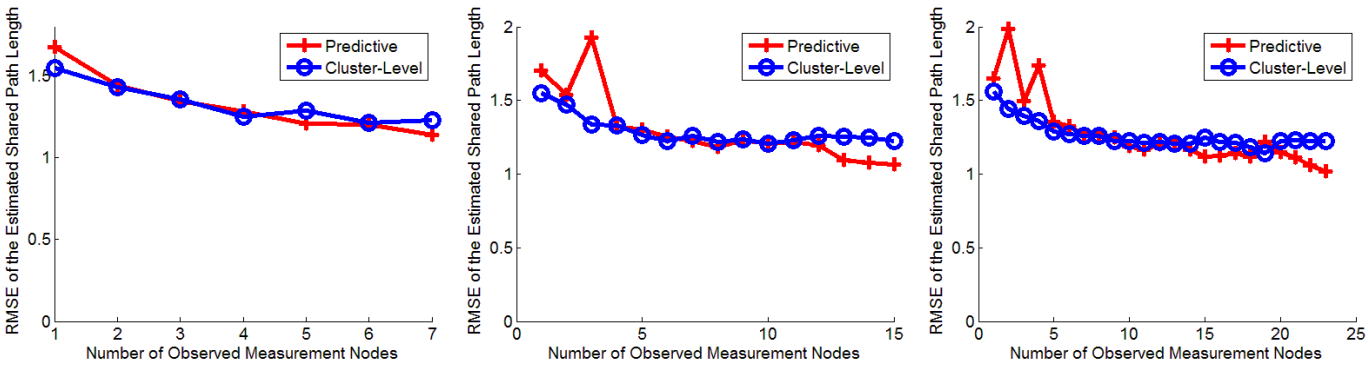

Figure 12: Topology estimation performance for two different estimation methods in a 1000 node synthetic topology, ((Left) $M = 8$, (Center) $M = 16$, (Right) $M = 24$)

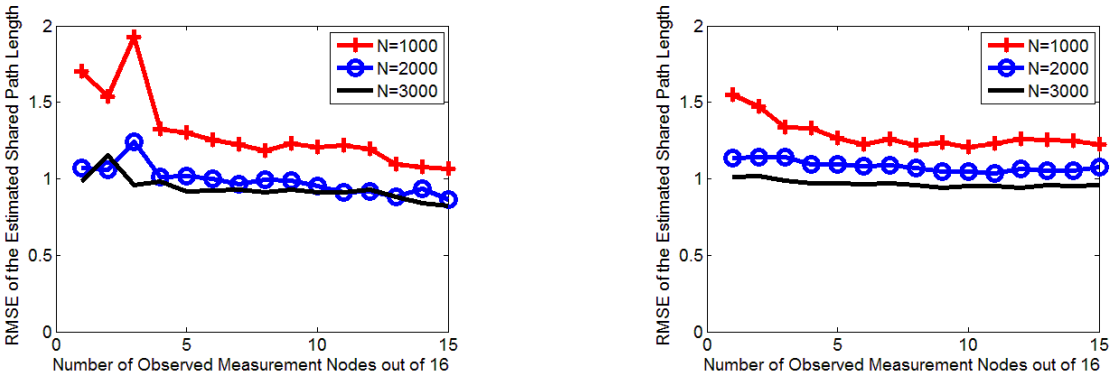

Figure 13: Performance of topology estimation algorithm in 1000, 2000, and 3000 node synthetic topologies with $M = 16$, (Left) - Predictive Function Topology Estimation, (Right) - Cluster-Level Topology Estimation.

8. REFERENCES

[1] The Honeynet Project. http://www.honeynet.org/, 2008.

[2] D. Alderson, L. Li, W. Willinger, and J. Doyle. Understanding Internet Topology: Principles, Models and Validation. *IEEE/ACM Transactions on Networking*, 13(6), December 2005.

[3] M. Bailey, E. Cooke, F. Jahanian, J. Nazario, and D. Watson. The Internet Motion Sensor: A Distributed Blackhole Monitoring System. In *Proceedings of The Network and Distributed Security Symposium (NDSS '05)*, San Diego, CA, January 2005.

[4] P. Barford, A. Bestavros, J. Byers, and M. Crovella. On the Marginal Utility of Network Topology Measurements. In *Proceedings of ACM Internet Measurement Workshop (IMW '01)*, San Francisco, CA, October 2001.

[5] CAIDA. The Skitter Project. http://www.caida.org/tools/measurement/skitter/, 2007.

[6] M. Coates, R.Castro, R. Nowak, M. Gadhiok, R. King, and Y. Tsang. Maximum Likelihood Network Topology Identification from Edge-based Unicast Measurements. In *Proceedings of ACM SIGMETRICS '02*, Marina Del Rey, CA, June 2002.

[7] F. Dabek, R. Cox, F. Kaashoek, and R. Morris. Vivaldi: A Decentralized Network Coordinate System. In *Proceedings of ACM SIGCOMM '04*, Portland, OR, August 2004.

[8] B. Eriksson, P. Barford, R. Nowak, and M. Crovella. Learning Network Topology from Passive Measurements. In *Proceedings of ACM Internet Measurements Conference (IMC) '07*, San Diego, CA, October 2007.

[9] M. Fomenkov, K. Keys, D. Moore, and K. Claffy. Longitudinal Study of Internet Traffic from 1998-2003. In *Proceedings of the Winter International Symposium on Information and Communications Technologies*, January 2003.

[10] C. Fraleigh, S. Moon, B. Lyles, C. Cotton, M. Khan, D. Moll, R. Rockell, T. Seely, and C. Diot. Packet-Level Traffic Measurements from the Sprint IP Backbone. *IEEE Network*, 17(6), Nov.-Dec. 2003.

[11] P. Francis, S. Jamin, C. Jin, Y. Jin, D. Raz, Y. Shavitt, and L. Zhang. IDMaps: A Global Internet Host Distance Estimation Service. *IEEE/ACM Transactions on Networking*, 9(5), 2001.

[12] P. Francis, S. Jamin, V. Paxson, D. Bryniewicz, and Y. Jin. An Architecture for a Global Internet Host Distance Estimation Service. In *Proceedings of IEEE INFOCOM '99*, New York, NY, April 1999.

[13] Z. Ghahramani and M. Jordan. Supervised Learning from Incomplete Data via the EM Approach. In

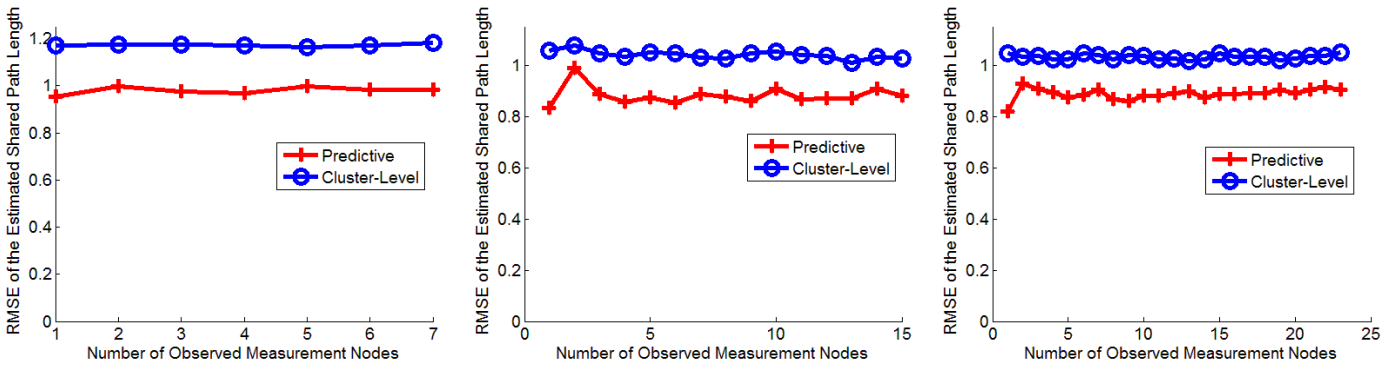

Figure 14: Topology estimation performance for two different estimation methods, with Gaussians mixture imputation in the Skitter topology. ((Left) $M = 8$, (Center) $M = 16$, (Right) $M = 24$)

Advances in Neural Information Processing, 1994.

[14] Z. Ghahramani and M. I. Jordan. Supervised Learning from Incomplete Data via the EM Approach. *Advances in Neural Information Processing Systems 6 (NIPS '94)*, 1994.

[15] R. Govindan and H. Tangmunarunkit. Heuristics for Internet Map Discovery. In *Proceedings of IEEE INFOCOM '00*, Tel Aviv, Israel, March 2000.

[16] B. Gueye, A. Ziviani, M. Crovella, and S. Fdida. Constraint-based Geolocation of Interenet Hosts. In *Proceedings of ACM Internet Measurement Conference (IMC '04)*, Taormina, Italy, October 2004.

[17] B. Gueye, A. Ziviani, M. Crovella, and S. Fdida. Constraint-Based Geolocation of Internet Hosts. *IEEE/ACM Transactions on Networking*, December 2006.

[18] C. Jin, H. Wang, and K. Shin. Hop-Count Filtering: An Effective Defense Against Spoofed Traffic. In *Proceedings of IEEE INFOCOM '03*, San Francisco, CA, April 2003.

[19] E. Katz-Bassett, J. John, A. Krishnamurthy, D. Weatherall, T. Anderson, and Y. Chawathe. Towards IP Geolocation Using Delay and Topology Measurements. In *Proceedings of ACM Internet Measurement Conference (IMC '06)*, Rio de Janerio, Brazil, October 2006.

[20] B. Krishnamurthy and J. Wang. On Network-Aware Clustering of Web Clients. In *Proceedings of ACM SIGCOMM '00*, Stockholm, Sweeden, August 2000.

[21] B. Lyon. The Opte Project. http://opte.org, January 2008.

[22] C. Faloutsos M. Faloutsos, P. Faloutsos. On Power-Law Relationships of Internet Topology. In *Proceedings of ACM SIGCOMM '99*, Cambridge, MA, August 1999.

[23] A.K. Jain M. Figueiredo. Unsupervised learning of finite mixture models. *IEEE Transaction on Pattern Analysis and Machine Intelligence*, 24:381–396, 2002.

[24] P. Madadevan, C. Hubble, D. Krioukov, B. Huffaker, and A. Vahdat. Orbis: Rescaling Degree Correlations to Generate Annotated Internet Topologies. In *Proceedings of ACM SIGCOMM '07*, Kyoto, Japan, August 2007.

[25] E. Ng and H. Zhang. Predicting Internet Network Distance with Coordinate-baseed Approaches. In *Proceedings of IEEE INFOCOM '02*, New York, NY, April 2002.

[26] Packetdesign. Route Explorer. http://www.packetdesign.com/, 2008.

[27] R. Pang, V. Yegneswaran, P. Barford, V. Paxson, and L. Peterson. Characteristics of Internet Background Radiation. In *Proceedings of ACM Internet Measurement Conference (IMC '04)*, Taormina, Italy, October 2004.

[28] N. Provos. A Virtual Honeypot Framework. In *Proceedings of the USENIX Security Symposium '04*, San Diego, CA, August 2004.

[29] Y. Shavitt and E. Shir. DIMES: Let the Internet Measure Itself. *ACM SIGCOMM Computer Communications Review*, 35(5), October 2005.

[30] Y. Shavitt and T. Tankel. Hyperbolic Embedding of Internet Graphs for Distance Estimation and Overlay Construction. *IEEE/ACM Transactions on Networking*, To Appear.

[31] N. Spring, R. Mahajan, and D. Wetherall. Measuring ISP Topologies with Rocketfuel. In *Proceedings of ACM SIGCOMM '02*, Pittsburgh, PA, August 2002.

[32] N. Spring, D. Wetherall, and T. Anderson. Reverse Engineering the Internet. In *Proceedings of Hotnets-II*, Cambridge, MA, November 2003.

[33] L. Tang and M. Crovella. Virtual Landmarks for the Internet. In *Proceedings of ACM Internet Measurement Conference (IMC '03)*, Miami, FL, October 2003.

[34] V. Yegneswaran, P. Barford, and D. Plonka. On the Design and Use of Internet Sinks for Network Abuse Monitoring. In *Proceedings of Recent Advances on Intrusion Detection (RAID '04)*, Sophia, France, September 2004.