Internet Iso-bar: A Scalable Overlay Distance Monitoring System

Yan Chen, and Randy H. Katz
University of California, Berkeley

Chris Overton,
Keynote Systems, Inc.

Abstract

Overlay network distance monitoring and estimation system can benefit many new applications and services, such as peer-to-peer overlay routing and location. However, there is a lack of such scalable system with small overhead, good usability, and good distance estimation accuracy and stability. Thus we propose a scalable overlay distance monitoring system, *Internet Iso-bar*, which clusters hosts based on the similarity of their perceived network distance, with no assumption about the underlying network topology. The centers of each cluster are then chosen as monitors to represent their clusters for probing and distance estimation. We compare it with other network distance estimation systems, such as Global Network Positioning (GNP) [1]. Internet Iso-bar is easy to implement and use, and has good scalability and small communication and computation cost for online monitoring. Preliminary evaluation on real Internet measurement data also shows that Internet Iso-bar has high prediction accuracy and stability. Finally, by adjusting the number of clusters, we can smoothly trade off the measurement and management cost for better distance estimation accuracy and stability.

1 Introduction

Given the rapid growth of the Internet, a new class of large-scale globally-distributed network services and applications have appeared. Examples include distributed content hosting services, overlay routing and location systems, application-level multicast, and peer-to-peer file sharing such as Napster and Gnutella. Because these systems have great flexibility in choosing their communication paths and targets, they can benefit significantly from dynamic network distance prediction in terms of metrics such as latency and bandwidth.

This paper aims to solve the following problem: Given N end hosts that may belong to several different administrative domains, how do we select a subset of them as monitoring sites and build an overlay distance estimation/monitoring service without knowing the underlying topology? The N end hosts could belong to a VPN of a globalized company or a peer-to-peer file sharing system such as Napster [2] and Gnutella [3]. To address the scalability problem, we propose a clustering scheme: *Internet Iso-bar*, which groups the end hosts together based on the similarity of distance perception. We then select a suitable host as a monitoring site for active and continuous probing.

We try several correlation metrics for clustering, and compare our scheme with other network distance estimation systems, such as Global Network Positioning (GNP) [1]. Internet Iso-bar is easy to implement and use, and has good scalability and small communication and computation cost for real-time monitoring. We further evaluate its estimation accuracy and stability with two sets of real Internet measurements. One is collected by the National Laboratory of Applied Network Research (NLANR) Active Measurement Project (AMP) [4]. The other is measured by Keynote Inc., the worldwide leader in Internet performance services [5]. These measurements enable us to measure the estimation stability on various time scales: daily, weekly, monthly and six-monthly. The results show that Internet Iso-bar has relatively high prediction accuracy and stability, and we expect to improve it further with better clustering methods. To the best of our knowledge, Internet Iso-bar is the first scalable overlay distance monitoring system and our work is the first attempt to evaluate the stability of various distance estimation schemes with real Internet measurement data.

The rest of the paper is organized as follows. In Section 2, we list our design goals and survey previous work. We present the clustering and distance estimation techniques of Internet Iso-bar in Section 3. Simulation methodology is given in Section 4, and stability and accuracy evaluation in Section 5. Finally, we conclude in Section 6.

2 Overview of Internet Iso-bar

2.1 Internet Iso-bar Goals

Scalability There are millions of users for peer-to-peer systems, such as Napster and Gnutella. So we cannot have any centralized point in the system to continuously measure distance to all hosts.

Small overhead and timely information Such monitoring system should provide real-time report for network distance between any two hosts. It requires small communication and computation costs for estimation computation.

Good distance estimation accuracy and stability While preserving scalability and timely distance estimation, it is desirable to have as accurate estimation as possible. Meanwhile, the system should perform stably as time evolves.

Incrementally deployable Our goal is to install the monitors on end hosts without interfering the existing core IP network. Furthermore, with more monitors deployed, the estimation accuracy should be incrementally improved.

Ease of use As a widely distributed system, we believe the monitoring service should be straightforward to use and requires no specific parameter tuning.

2.2 Alternative Architectures and Related Works

There are numerous work on Internet end-to-end distance estimation/monitoring systems, both in research prototypes and commercially deployed networks.

Content Distribution Network (Akamai's Network Operations Command Center (NOCC)) For each client address prefix (subnets), Akamai uses traceroute from all CDN servers to figure out that there are a few core-routers (close to the clients) that are always on the path to the client clusters. Then they measure the distance from each Internet Data Center (IDC) which hosts CDN edge servers to these routers to decide the relative distance to these clients. The map is periodically updated (i.e., every 20 seconds) [6]. Although it works fine in real operation, this approach has a potential scalability problem with regards to the number of IDCs in Internet Service Providers (ISPs) and the number of client groups. There are more than 8800 ISPs these days. Assuming that the clients are grouped by autonomous systems (AS's), there are 20,000 AS's, leading to 176M traceroute measurements for building the distance map. If more accurate client clustering techniques suggested in [7] are used, the scalability problem worsens because the number of client clusters reaches the order of hundreds of thousands.

IDMaps In the pioneering work of Francis *et al.* [8, 9], the authors examined the network distance prediction problem from a topological point of view and proposed the first complete solution called IDMaps. IDMaps is an infrastructure service in which special HOPS servers (called *Tracers*) measure the distance among themselves and to other end hosts. The distance between two end hosts A and B is estimated as the sum of 3 distances: distance between A and its nearest Tracer T_1, distance between B and its nearest Tracer T_2 and the shortest path distance between T_1 and T_2. To achieve scalability, they group the clients by Address Prefix (AP) which is a consecutive address range of IP addresses within which all hosts are equidistant (with some tolerance) to the rest of Internet. Grouping like that leads to hundreds of thousands of APs, which are further clustered based on the network proximity. The problem is that such clustering requires full global network topology. As a heuristic, Tracers are placed on transit autonomous systems (AS). However, this requires the cooperation of network providers. In contrast, the monitors of Internet Iso-bar are placed on overlay nodes, avoiding the need for help from network providers. IDMaps also faces problems with prediction accuracy. Firstly, the estimation based on triangulation inequality doesn't hold unless the Tracers are always placed on or very close to the shortest path between the client groups. Secondly, the proximity-based clustering is not as accurate as the similarity-based clustering used by Internet Iso-bar as we will show in Section 5.

Network Distance Maps To tackle the measurement scalability problem, Theilmann and Rothermel have proposed *network distance maps* [10]. Assuming the existence of measurement servers (mServers), a hierarchical tree of mServers can be constructed so that each mServer only measures the distance to its siblings. Each host is then assigned to its closest mServer. The distance between host A and B is estimated by the distance between each of their ancestor mServers. This hierarchical clustering of monitors is complementary to the monitor site selection scheme of Internet Iso-bar. We thus omit it for comparison in Table 1.

GNP Recently, *Global Network Positioning (GNP)* has been proposed by Ng and Zhang [1]. It is based on absolute coordinates computed from modeling the Internet as a geometric space. Each end host maintains its own coordinates and network distances to other hosts are predicted by evaluating a *distance function* (e.g., Euclidean distance) over their coordinates. Evaluation based on real Internet measurement data shows it is much more accurate than IDMaps. However, as they admit, the landmark sites are potential bottlenecks because every host has to measure its distance to the landmarks to compute and update its coordinates. Moreover, it is hard to achieve real-time distance estimation – both source and destination have to obtain measurements to $D + 1$ landmarks (D being the dimension used in estimation), recompute the coordinates, and exchange the coordinates to get the estimation. Thus it is not suitable as an online monitoring system. Moreover, GNP can only model symmetric distance, and its optimization algorithms are expensive to run.

RON A *Resilient Overlay Network (RON)* [11] is an architecture that allows distributed Internet applications

Properties	Akamai NOCC	IDMaps	GNP	RON	Internet Iso-bar
Scalability	Traceroute from each edge server farm to all client subnets	Proximity-based clustering of APs, $O(C^2 + AP)$ measurements	NK measurements (each landmark takes $O(N)$ of them)	N^2 measurements	$C^2 + N$ measurements
Estimation technique and accuracy	Use distance between edge server farms to nearby router of clients	Based on triangulation inequality and proximity-based clustering, inaccurate	Based on high-dimension coordinates, accurate	Exact measurements	Similarity-based clustering, accurate
Timely monitoring	Yes	Yes	No	Yes	Yes
Monitors/landmarks deployment	CDN edge servers	Transit AS's	End hosts	End hosts	End hosts

Table 1: Comparison of various Internet distance estimation systems, assuming there are N end hosts, AP address prefixes, K landmarks and C clusters

to detect and recover from path outages and periods of degraded performance within several seconds. They keep monitoring the quality of Internet paths between *every* pair of RON nodes. Although the scheme achieves high accuracy, it fails to scale.

Table 1 compares these measurement schemes with Internet Iso-bar, which will be addressed in the next section.

3 Internet Iso-bar

The key idea of *Internet Iso-bar* as follows: First, we use various correlation metrics (Section 3.1) to determine the *correlation distance* between pairs of hosts (denoted as $cor\_dist(*,*)$). Then we apply generic clustering methods to cluster the N hosts into K clusters, and choose the center of each cluster as the monitor for that cluster (Section 3.2). Then the monitors continuously measure the distance among themselves as well as to the hosts in its cluster to compute the distance prediction for any pair of hosts (Section 3.3).

As will be demonstrated by our experiments (Section 4), *Internet Iso-bar* is very flexible to provide distance monitoring for both peer-to-peer and client-server types of systems. The former requires distance monitoring for every pair of peer hosts, while the latter only needs distance between every (client, server) pair. Figure 1 shows the Internet Iso-bar architecture for a peer-to-peer system. Some correlation distance measurements require *landmark sites*, which will be explained in Section 3.1.

3.1 Correlation Distance

In this section, we describe the correlation distance metrics we use. We explore two orthogonal metrics: one based on network distance correlation, and the other based on geographical distance correlation.

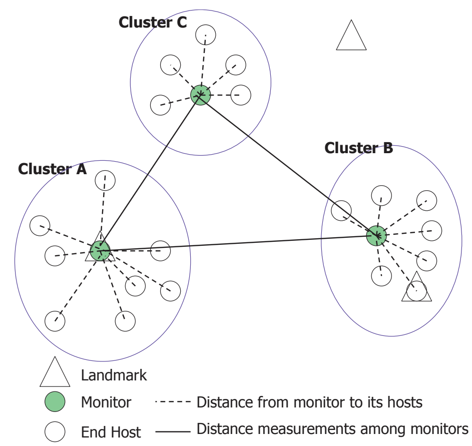

Figure 1: Internet Iso-bar architecture for a peer-to-peer system

3.1.1 Clustering based on network distance correlation: Distances in the Internet can be of multiple metrics, such as number of hops, latency, loss rate, jitter, bandwidth, etc. In particular, we study the ICMP round trip time (RTT) and TCP initial connection time in our paper and we denote the measured network distance between host i and host j as $net\_dist(i, j)$. The simplest form is to use $net\_dist(i, j)$ for clustering directly.

Using proximity:

$$cor\_dist(i, j) = net\_dist(i, j)$$

This is adopted by IDMaps [9] and Network Distance Maps [10]. However, it requires full network topology (N^2 distance) and evaluation in Section 5 proves that is not very accurate. Next, we propose clustering based on network similarity. To that end, we introduce the the concept of landmarks, which is different from that in GNP. Each host i measure distance to landmarks and form *network distance vector* (referred as $netV_i$). $netV_i$

is used for clustering purposes only and not used for distance estimation like in GNP. $NetV_i$ is a m-dimensional vector where m is the number of landmark hosts. In our notation, element k (referred as w_k, $1 \leq k \leq m$) of $netV_i$ is $net\_dist(i,k)$ where k is the kth landmark host. The landmarks can be all or a subset of the end hosts plus any outside servers that accepts measurements.

Using the Euclidean distance between network distance vectors:

$$cor\_dist(i,j) = |netV_i - netV_j|$$

$$= \sqrt{\sum_{k=1}^{m}(w_k(netV_i) - w_k(netV_j))^2}$$

Using the vector similarity between network distance vectors: Cosine vector similarity has been widely applied to measure the similarity between two vectors. Here we use its complement as the correlation distance.

$$cor\_dist(i,j) = 1 - \frac{netV_i \cdot netV_j}{|netV_i||netV_j|} =$$

$$1 - \frac{\sum_{k=1}^{m} w_k(netV_i) \cdot w_k(netV_j)}{\sqrt{\sum_{k=1}^{m}[w_k(netV_i)]^2 \cdot \sum_{k=1}^{m}[w_k(netV_j)]^2}}$$

Essentially, if we view each vector as an arrow in high-dimension space, the formula above gives the sine of the angle between the arrows of $netV_i$ and $netV_j$.

3.1.2 Clustering based on geographical distance correlation: Given the longitude and latitude of each host, the correlation distance can be defined as: $cor\_dist(i,j) =$

$$\sqrt{(longitude_i - longitude_j)^2 + (latitude_i - latitude_j)^2}$$

3.2 Generic Clustering Methods

We define the radius of cluster i as the maximum distance between the monitor (center node) and any hosts in cluster i (C_i). Our clustering seeks to minimize the maximum radius of all clusters. The placement of a given number of monitors such that the maximum distance from a host to its nearest monitor is minimized, is known as minimum K-center problem [12]. We use the algorithm in [13] to achieve an approximation within a factor of 2 in $O(N^3)$ time (N is the number of end hosts).

In addition, we explore the dual problem of limiting the radius of each cluster and minimizing the number of clusters. It can be converted to minimum set cover problem, i.e, given graph G with N hosts and a distance bound d, find a smallest subset of N' such that the distance between any host h and its "center" host c_h is bounded by d. More formally, find the minimum K, such that there is a set $N' \subset N$ with $|N'| = K$ and $\forall\ h \in N$, $\exists\ c_h \in N'$ such that distance(h, c_h) $\leq d$. The problem above is NP-hard. Grossman and Wool [14] recommend using greedy approximation algorithm than other methods because the former gives comparable results at a fraction of the time ($O(N^2)$ complexity). The approximation ratio is $\ln|N|$ [14]. In the rest of the paper, we refer to the first method as *limit_number clustering* and the second one as *limit_diameter clustering*.

3.3 Predicted Distance Computation

For a peer-to-peer system, we predict the distance between any pair of hosts host i and j. If they belong to the same cluster with monitor m, $predicted\_dist(i,j) = (dist(i,m) + dist(j,m))/\ 2$. Otherwise, assume that the monitor for i is m_i and for j is m_j, $predicted\_dist(i,j) = dist(m_i, m_j)$. Note that usually the relative number of inter-cluster estimates is much higher than the number of intra-cluster estimates. For example, given N hosts evenly divided into K clusters, and considering all pairs of hosts for distance estimation, the ratio of inter-estimation estimates vs. intra-estimation estimates will be approximately $K-1$.

For a client-server system, we predict the distance between any client c and server s. Assume that the monitor for c is m, $predicted\_dist(c,s) = dist(m,s)$.

Because there is delay for the monitors to send the predicted distance to the requesting hosts, we need to estimate the next measurements of monitors based on their past measurements, then use the estimation to compute the predicted distance of requesting hosts, such as $predicted\_dist(i,j)$ and $predicted\_dist(c,s)$ above.

There are numerous techniques for estimating the next measurements of monitors. The simplest way is to just use the last observation. More precisely, we can use the aggregated information from the sliding window analysis as shown below:

1. **Weighted sum of the measurements in the window:** The weight is assigned in linear proportion to the order of the measurement in the sliding window. That is, the weight of the ith measurement in sliding window of size n is $\frac{i}{\sum_{k=1}^{n} k}$.
2. **Statistics of the measurements in the window:** As the Internet distance measurement are heavy-tailed distribution (revealed in [15], we also verified it with our measurement data), we use the geometric mean of the sliding window as prediction.

We have tried both methods using three window sizes: 5, 15 and 25. We find that the geometric mean performs slightly better than the weighted sum, while window size of 15 and 25 outperforms window size of 5 a little. So in the rest of the paper, we use the geometric mean of sliding windows with size 15 to aggregate past measurements.

4 Evaluation Methodology

4.1 Internet Measurement Data

The performance of the different methods for estimating latency is evaluated with two sets of real Internet measurement data: the National Laboratory of Applied Network Research (NLANR) Active Measurement Project (AMP) data [4] and Keynote Web Site Perspective measurement data [16].

4.1.1 NLANR AMP Data: There are 119 sites participating in the AMP project, each of which uses a dedicated machine to collect measurement data. Each measurement site uses ICMP messages to collect the Round Trip Time (RTT) between itself and every other site participating in this project. Measurements are made every minute, resulting in a total of 1440 measurements for each day. We collected one month of raw data (6/25/01 to 7/24/01), and one day of relatively current data (12/6/01). The 119 sites were filtered to 106 due to some sites either 1) not collecting a full day's worth of data or 2) not making measurements to all other participating sites. For this case, the number of hosts, N is 106 while the number of landmarks used for testing varied as will be shown in Figure 3.

This data will serve two purposes. First, for one day we calculate the geometric mean of the measurements between each pair of sites i and j and use that aggregated value as $net\_dist(i, j)$ in Section 3 for clustering or coordinate calculation (GNP). We refer to that day as the *birth date*. Second, we use the raw data (i.e., 1440 measurements) of another day to evaluate the accuracy of estimation resulting from each technique. We refer to it as the *estimation date*.

4.1.2 Keynote Web Site Perspective Data: Keynote measurement agents [16] measure Web site performance on a worldwide infrastructure of public measurement computers in over 100 statistically selected locations in over 50 cities in the world. While almost all NLANR AMP probes have dedicated T1 connections to the Abilene backbone network [17], the Keynote *Business Perspective* agents used for this study have T1 or greater connectivity to representative major ISPs in many major cities in the world. We used the *Keynote Business 40* index, consisting mainly of front pages for 40 of the most popular web servers used by US businesses. The 102 agents then measure the TCP connection time to each of the 40 web sites every 15 minutes. For this study of performance, we filtered out the occasional error measurements due to such causes as DNS lookup and TCP timeouts. We collected one month of raw data from 11/13/2001 to 12/13/2001. For this case, the number of hosts, N, is 102 while the number of landmarks, M, that we used for testing is 40.

These measurement data are used in a way similar to the NLANR AMP data discussed before. We cluster the agents based on the similarity of their distance to the 40 Web servers and choose monitors from each cluster as representatives. However, because the distances between the agents were not given, we cannot select landmarks or compute their coordinates for GNP approach. Neither can we cluster the agents based on network proximity or geographical proximity due to the lack of information.

The advantage of this approach lies in using more typical web traffic that is the case for peer-to-peer ICMP messages.

4.2 Estimation Accuracy Quantification

To measure how well an estimated distance compares with actual network distance, we use the following relative prediction error metric defined in [18]. Here E refers to probabilistic expectation.

$$relative\ error = E[|log_2(\frac{predicted\ distance}{measured\ distance})|]$$

The expectation is computed over each of the measurement events (i.e., 1440 in the NLANR daily traces and 144 in the Keynote daily traces) and reflects the ratio by which the estimator typically misses the target. Given different estimation techniques, the estimated distance could be either *static* or *dynamic* as explained below.

GNP is not scalable enough for online update of the coordinates. Therefore we use each site' coordinates obtained from the birth date for distance estimation. That is, for any pair of hosts i and j, the same estimated value is used consistently throughout the daily relative error calculation.

For clustering-based approaches, we use *active monitoring* (in other words, *dynamic estimation*) in which monitors run the active probing. Then we use various methods in Section 3.3 for prediction.

4.3 Analysis of Estimation Accuracy

4.3.1 Static analysis: For static analysis, we use the measurement data of the same day both for offline setup (clustering or coordinates computation) and for online estimation. That is, the birth date and the estimation date are the same. This will help give a sense of the absolute accuracy of each estimation technique. It can also be used as a sanity check.

4.3.2 Stability analysis: Of greater interest is how well estimations derived from a single day's data perform over multiple time intervals. The time intervals we will examine in our stability analysis are one day, three days, one week, two weeks, one month, and six months from the birth date. The birth date is 6/25/01 in all of the intervals examined.

5 Evaluation

In this section, we start with the list of Internet distance estimation techniques we evaluated. For the time being, we only implement the clustering for optimizing the worst prediction accuracy in Section 3.2. So we compare the two generic clustering methods in that category and evaluate the sensitivity of Internet Iso-bar to different number of landmarks. Finally, we present the results of accuracy and stability analysis of various estimation schemes.

5.1 Internet Distance Estimation Techniques Evaluated

We compare the accuracy and stability of six distance estimation schemes listed below:

- Omniscient approach (Omni)
- GNP
- Clustering based on Euclidian distance of network distance vector (*Net_sim*)
- Clustering based on vector similarity of network distance vector (*Net_vsim*)
- Clustering based on geographical proximity (*Geo_prox*)
- Clustering based on network proximity (*Net_prox*)

The omniscient approach can be characterized as a naive approach, with prediction using complete knowledge of the distances between all pairs of sites on the birth date. In this approach, the distance estimation between any source and destination is is simply the geometric-mean of the actual measured distance on the birth date. For a fair comparison, we use 15 landmarks (in 7 dimensions as configured in [1]) for GNP and 15 clusters for clustering approaches.

5.2 Evaluation of Generic Clustering Techniques

We compare limit_number clustering and limit_diameter clustering under four correlation distance metrics defined in Section 3.1. Figure 2 shows the static and stability analysis results from NLANR data. They all have similar results, with limit_number clustering performing a little better than limit_radius clustering. Similar trends are also visible when we use proximity-based correlation distance for clustering. Thus, for the rest of the paper, we only use limit_number clustering for evaluation purposes.

Also when we compare clustering based on *Net_sim* and *Net_vsim*, the results are also quite similar, but *Net_sim* slightly outperforms *Net_vsim*. Thus we only use *Net_sim* for the rest of stability comparison.

Similar trends also exist in the evaluation of Keynote data (Figure 6).

5.3 Sensitivity of Internet Iso-bar to different number of landmarks

In this section, we try to answer this question: can a small random subset of landmarks capture network similarity enough for a good clustering? We compare the performance of *Net_sim* under NLANR data with different number of landmarks M, with M being 6, 15, 60 and 106. The landmarks chosen in each case are totally random. Figure 3 shows that they do not differ much in terms of prediction accuracy. We can get performance close to that of using all sites as landmarks even by using a very small number of landmarks. This suggests that Internet Iso-bar can be very efficient in computation.

5.4 Stability Results of Prediction Accuracy

Figures 4 and 6 show the cumulative distribution function (CDF) of the relative prediction errors for both static and stability analysis. For NLANR data (Figure 4), we test the stability of various distance estimation schemes over a six month period. The birth date for estimation is 06/25/01, and the estimation dates are 6/25/01, 6/26/01, 6/28/01, 7/01/01, 7/8/01, 7/15/01, 7/24/01, and 12/06/01 respectively. The figures in this section are results for tests with the number of landmarks, M, being 106.

The stability of *Omni* and various networking similarity based clustering approaches are also evaluated with Keynote data over a one month period (Figure 6). The birth date is 11/13/01, and the estimation dates are 11/13/01, 11/14/01, 11/16/01, 11/20/01, 11/27/01 and 12/13/01 respectively.

We make the following observations. First, estimation based on the omniscient approach gives the highest accuracy on all the estimation dates. We believe this represents an underlying stability in the current Internet. Also note that as in any static estimation scheme, the omniscient approach can not report any transient network congestion. Furthermore, it requires the full n by n network distance matrix, and thus not scalable. It is worth pointing out that the omniscient approach does not achieve perfect accuracy when it is used to estimate distance on the actual birth date. This is because the distance estimate used for a pair of hosts is the geomet-

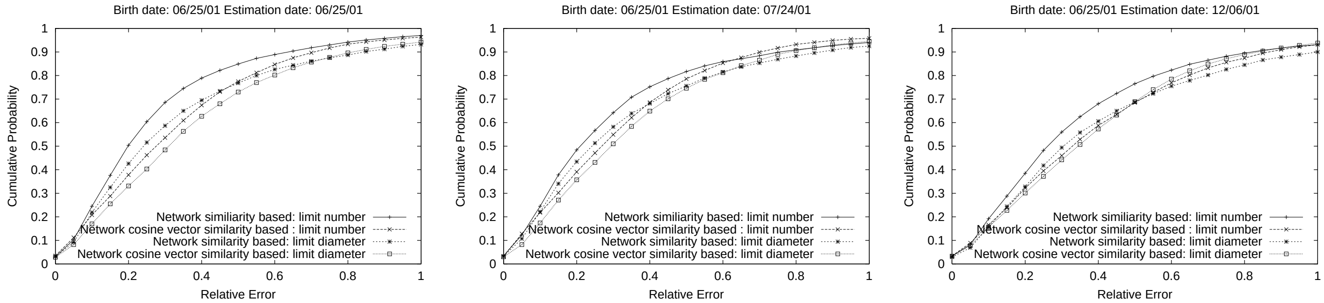

Figure 2: Evaluation of generic clustering methods with static analysis (left) and stability analysis (middle, right) of NLANR data

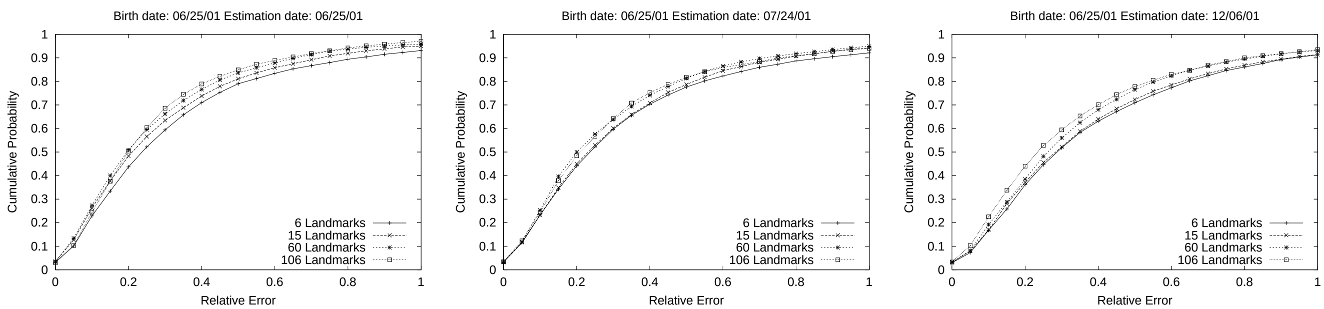

Figure 3: Sensitivity of Internet Iso-bar to various number of landmarks: static analysis (left) and stability analysis (middle, right) of NLANR data

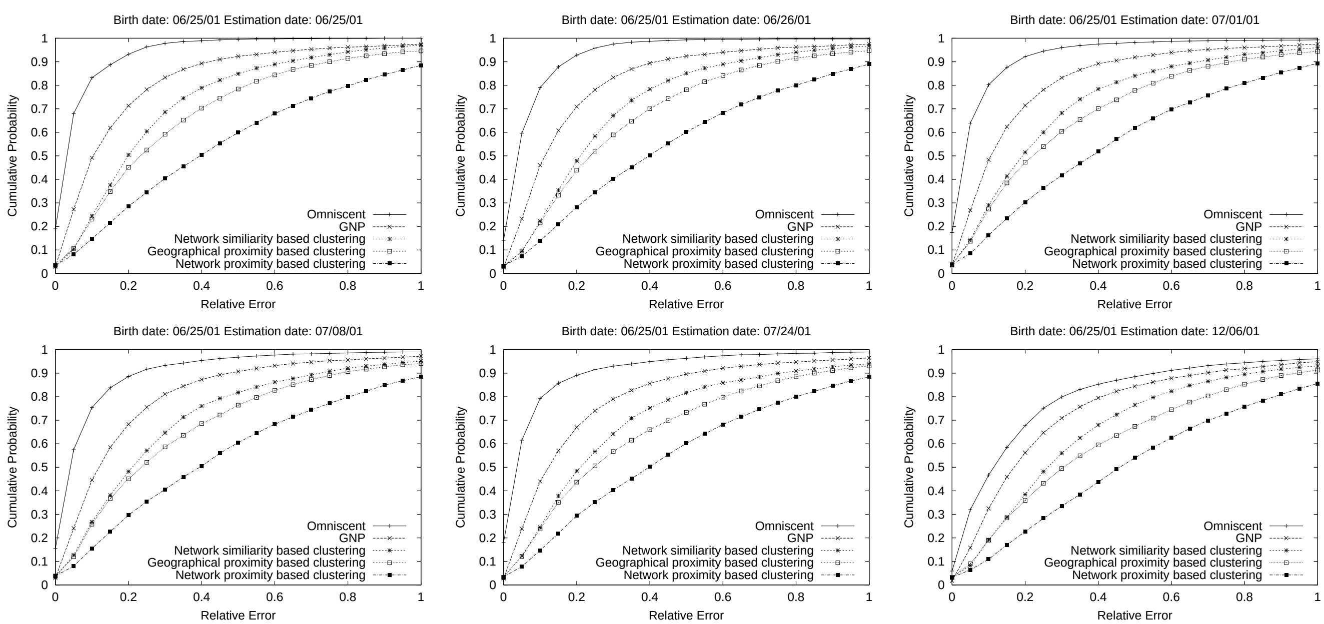

Figure 4: Cumulative Distribution Function (CDF) of relative prediction errors for both static and stability analysis (five different time intervals) with various distance estimation services, evaluated with NLANR data

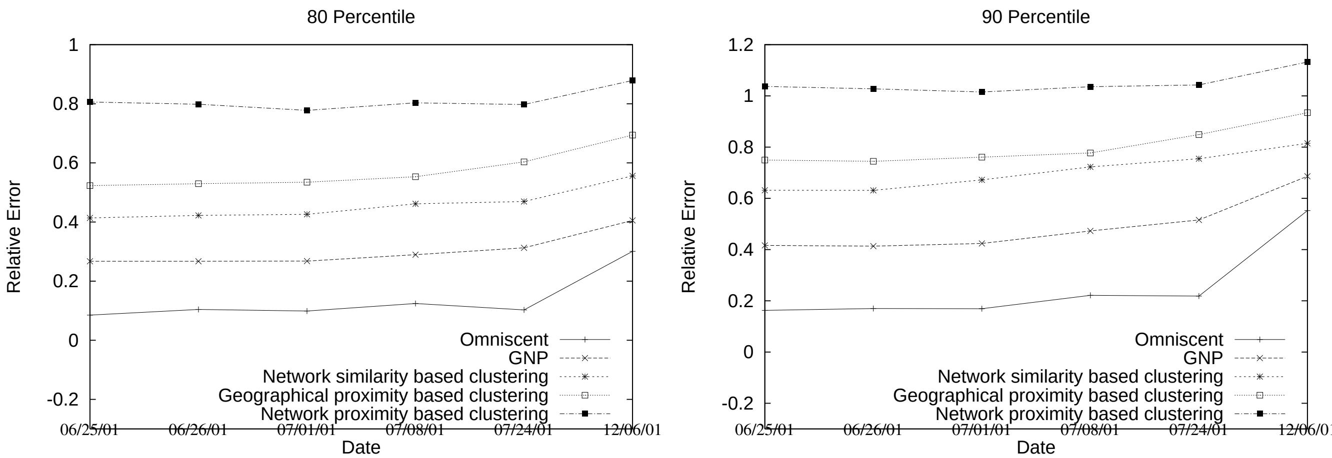

Figure 5: 80 percentile (left) and 90 percentile (right) of the relative errors for various distance estimation services under six different time intervals, evaluated with NLANR data

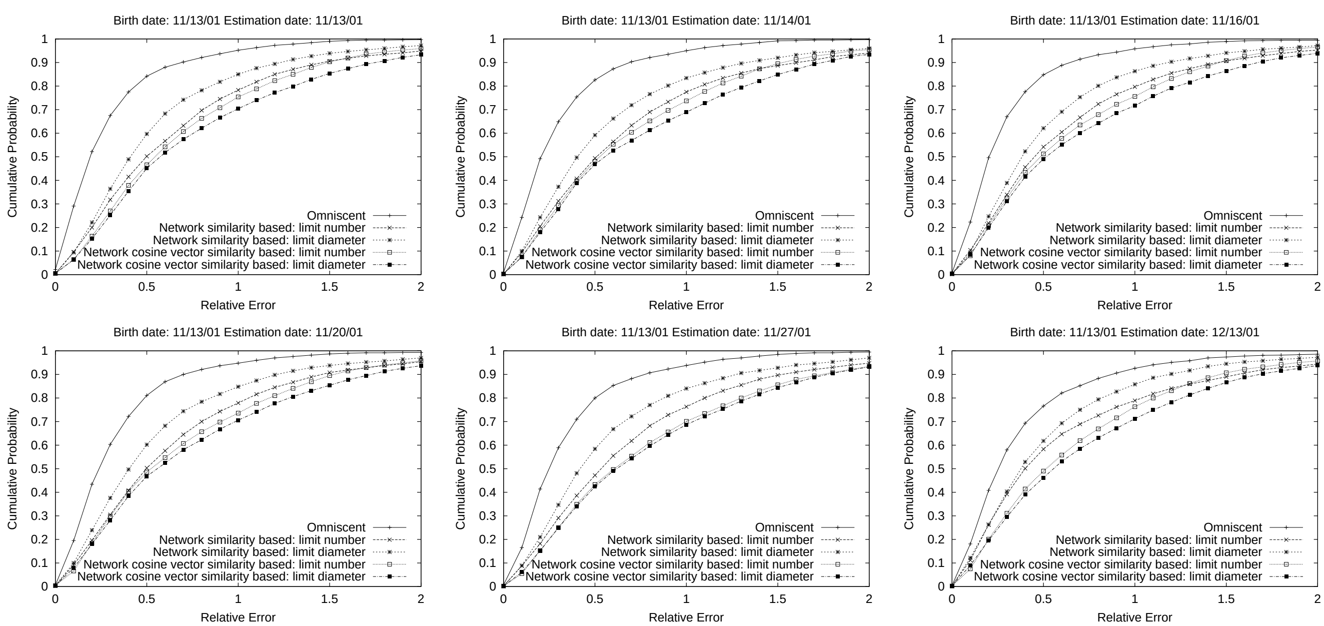

Figure 6: Cumulative Distribution Function (CDF) of relative prediction errors for both static and stability analysis (five different time intervals) with various distance estimation services, evaluated with Keynote data

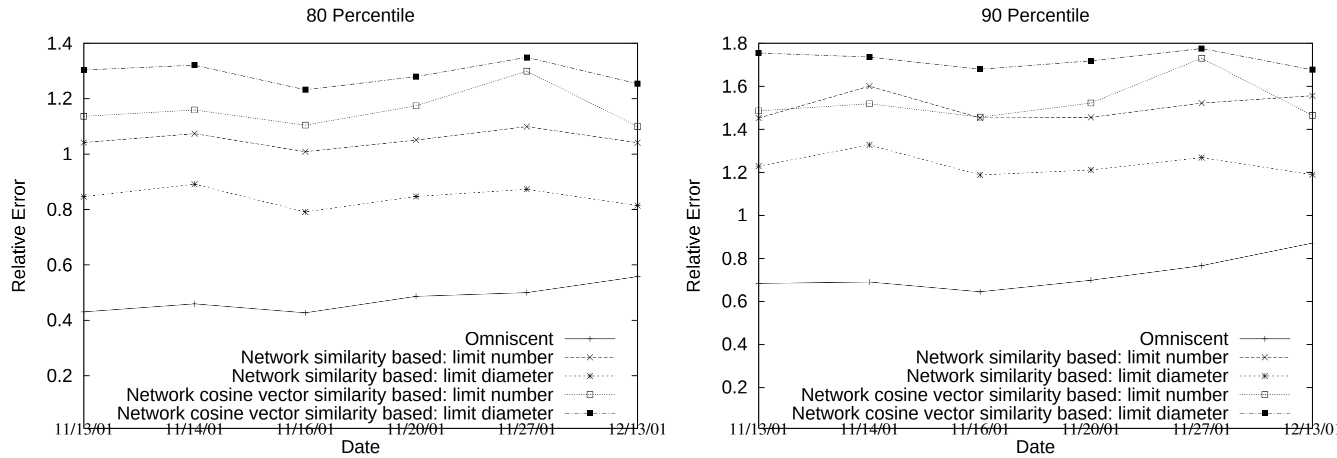

Figure 7: 80 percentile (left) and 90 percentile (right) of the relative errors for various distance estimation services under six different time intervals, evaluated with Keynote data

ric mean of all RTT samples obtained during the course of the whole day. The error reflects the amount of fluctuation in RTT within a day.

Second, GNP is the second best performer. This implies that GNP provides a very accurate estimate when the underlying data set is stable. However, the accuracy difference between GNP and Internet Iso-bar is almost negligible. Take the results of 06/25/01 for instance, the ratio between predicted distance and real distance for GNP is $2^{0.267} = 1.2$, while for Net_sim is $2^{0.4} = 1.32$. Given that 90% geometric mean of pair-wise RTT is less than 91 ms for NLANR (06/25/01), the prediction difference between GNP and Net_sim is less than 10 ms. In addition, GNP requires all hosts to constantly measure distance to the landmarks for online monitoring, which is impractical. Thus it cannot report timely congestion/loss, but that information is more interesting to many clients.

Third, the network distance similarity based clustering yields the highest accuracy among all the clustering based approaches. It performs much better than the network proximity based clustering, which is used in IDMaps [12] and Network Distance Maps [10].

Finally, it is interesting to see that geographical distance proximity based clustering performs better than network distance proximity based clustering. This may be because most sites in our experimental data set are educational and research institutes, and they are connected by the same backbone: Internet2. The correlation between geographical distance and network distance need to be further verified using the Keynote data. Previous work has indicated that estimation techniques based on geographic distance in general result in poor accuracy [1].

Figures 5 and 7 show the amount of relative errors that are below the 80th percentiles in the CDF of relative errors, and how the relative errors change over time. In this metric, a lower value represents a high level of accuracy. Most methods are relatively stable except omniscient scheme.

6 Conclusion

In this project we propose the framework for a novel clustering-based overlay monitoring service, the *Internet Iso-bar*, which clusters hosts together based on the similarity of their perceived distance to the rest of the Internet. Compared with the state-of-the-art work, GNP, Internet Iso-bar has much better scalability and smaller measurement overhead for online monitoring without much loss in estimation accuracy and stability.

There is much exciting future work. Firstly, we will try other clustering methods to improve the overall estimation accuracy and do more complete test with Keynote data. Secondly, we will design and test the dynamic service model for Internet Iso-bar, i.e., how to adapt to the dynamic joining and leaving of the clients. One simple way could have the joining client to ping the landmark sites to get its network distance vector. Then it will collect the distance vectors of the monitoring sites and choose the one with best correlation to join. We may also need to dynamically set up or turn off monitors, i.e., split or merge of clusters. Lastly, we want to examine more metrics beyond latency prediction, e.g., the congestion/loss prediction in term of true positive and false positive ratios.

7 Acknowledgments

We graciously acknowledge sponsorship and grants from DARPA (grant N66061-99-2-8913), California Micro Grant #01-042, Ericsson, Nokia, Siemens, Sprint, NTTDoCoMo and HRL laboratories. We thank Chris Karlof and Yaping Li for discussion and help with implementation. We thank Vern Paxon and the anonymous reviewers for their valuable suggestions. Tony McGregor kindly provides us the NLANR AMP data to make the study possible.

References

[1] T. Ng and H. Zhang, "Predicting Internet network distance with coordinates-based approaches," in *to appear in Proceedings of IEEE Infocom*, 2002.

[2] S. Fanning, "Napster," `http://www.napster.com`.

[3] Gnutella, "`http://www.gnutellanews.com/information/what_is_gnutella.shtml`," .

[4] NLANR, "http://amp.nlanr.net/," .

[5] Keynote Inc., "http://www.keynote.com/," .

[6] P. Danzig, ," Former V. P. Technology of Akamai, Personal communication.

[7] B. Krishnamurthy and J. Wang, "On network-aware clustering of web clients," in *Proc. of SIGCOMM '2000*.

[8] P. Francis et al., "An architecture for a global Internet host distance estimation service," in *Proceedings of IEEE Infocom*, 1999.

[9] S. Jamin et al., "On the placement of Internet instrumentation," in *Proceedings of IEEE Infocom*, 2000.

[10] W. Theilmann and K. Rothermel, "Dynamic distance maps of Internet," in *Proceedings of IEEE Infocom*, 2000.

[11] D. G. Andersen, H. Balakrishnan, M. F. Kaashoek, and R. Morris, "Resilient overlay networks," in *Proc. of ACM SOSP*, 2001.

[12] P. Francis et al., "Idmaps: A global internet host distance estimation service," *IEEE/ACM Trans. on Networking*, Oct. 2001.

[13] V. Varirani, *Approximation Methods*, Springer-Verlag, 1999.

[14] T. Grossman and A. Wool, "Computational experience with approximation algorithms for the set covering problem," *Euro. J. Operational Research*, vol. 101, no. 1, pp. 81–92, August 1997.

[15] A. Medina, I. Matta, and J. Byers, "On the origin of power laws in Internet topologies," in *ACM Computer Communication Review*, April 2000.

[16] Keynote Inc., "Assure the quality of your Web site performance," http://www.keynote.com/solutions/html/web_site_perspective.html.

[17] Abilene Network Backbone, "`http://www.ucaid.edu/abilene/home.html`," .

[18] Y. Zhang, N Duffield, V. Paxson, and S. Shenker, "On the constancy of Internet path properties," in *Proceedings of SIGCOMM Internet Measurement Workshop 2001*.