Constructing Internet Coordinate System Based on Delay Measurement *

Hyuk Lim†
School of Electrical Engineering and Computer Science
Seoul National University
Seoul 151-744, Korea
hyuklim@csl.snu.ac.kr

Jennifer C. Hou
Department of Computer Science
University of Illinois at Urbana-Champaign
1304 W. Springfield Avenue
Urbana, IL 61801
jhou@cs.uiuc.edu

Chong-Ho Choi
School of Electrical Engineering and Computer Science
Seoul National University
Seoul 151-744, Korea
chchoi@csl.snu.ac.kr

ABSTRACT

In this paper, we consider the problem of how to represent the locations of Internet hosts in a Cartesian coordinate system to facilitate estimate of the network distance between two arbitrary Internet hosts. We envision an infrastructure that consists of beacon nodes and provides the service of estimating network distance between two hosts without direct delay measurement. We show that the principal component analysis (PCA) technique can effectively extract topological information from delay measurements between beacon hosts. Based on PCA, we devise a transformation method that projects the distance data space into a new coordinate system of (much) smaller dimensions. The transformation retains as much topological information as possible and yet enables end hosts to easily determine their locations in the coordinate system. The resulting new coordinate system is termed as the *Internet Coordinate System (ICS)*. As compared to existing work (e.g., IDMaps [1] and GNP [2]), ICS incurs smaller computation overhead in calculating the coordinates of hosts and smaller measurement overhead (required for end hosts to measure their distances to beacon hosts). Finally, we show via experimentation with real-life data sets that ICS is robust and accurate, regardless of the number of beacon nodes (as long as it exceeds certain threshold) and the complexity of network topology.

*This work was supported in part by the Brain Korea 21 Information Technology Program.

†H. Lim is currently a visiting student in the Department of Computer Science at University of Illinois at Urbana Champaign.

IMC'03, October 27–29, 2003, Miami Beach, Florida, USA.

Categories and Subject Descriptors

C.2.5 [**Local and Wide-Area Networks**]: Internet; C.4 [**Performance of Systems**]: Measurement techniques

General Terms

Algorithm, Measurement

Keywords

Internet distance service, principal component analysis, coordinate system.

1. INTRODUCTION

Discovery of Internet topology has many advantages for design and deployment of topology sensitive network services and applications, such as nearby server selection, overlay network construction, routing path construction, and peer-to-peer computing. The knowledge of network topology enables each host in these systems to make better decisions by exploiting its topological relations with other hosts. For example, in peer-to-peer file sharing services such as *Napster*, *Gnutellar*, and *eDonkey*, a client can download shared files from a peer that is closer to itself, if the topology information is available. Among several categories of approaches to infer network topology, the measurement based approach may be the most promising, whereby the network topology is constructed by end-to-end measurement of network properties, such as bandwidth, round-trip time, and packet loss rate. In this paper, we focus on topology construction based on end-to-end delay (round-trip time) measurement, and use the term "network distance" for the round-trip time between two hosts.

The primary goal of constructing network topology is to enable estimation of the network distance between arbitrary hosts without direct measurement between these hosts. Several approaches have been proposed among which IDMaps [1] and GNP [2] may have received the most attention. Both assume a common architecture that consists of a small number of well-positioned infrastructure nodes (called *beacon nodes* in this paper). Every beacon node measures its distances to all the other beacon nodes and uses these measurement results to infer the network topology. A host estimates its distance to the other ordinary hosts by measuring its distances to beacon nodes (rather than to the other hosts). A host benefits from using this architecture, as it needs only to perform a small number of measurements and will be able to infer its network distance to a large number of hosts (such as servers).

One important issue in realizing these measurement architectures is how to represent the location of a host. IDMaps and Hotz's triangulation [3, 4], for example, uses the original distances to beacon nodes to represent the location of a host, while GNP [2] transforms the original distance data space into a Cartesian coordinate system and uses coordinates in the coordinate system to represent the location. As will be discussed in Section 3, the major advantage of representing network distances in a coordinate system is that it enables extraction of topological information from the measured network distance data. As a result, the accuracy in estimating the distance between two arbitrary hosts will be improved especially in the case that the number of beacon nodes is small. To construct a new coordinate system, GNP formulates an optimization problem that minimizes the difference between the measured network distance in the distance data space and the Euclidean distance in a Cartesian coordinate system, and applies the Simplex Downhill method to solve the minimization problem. In spite of its many advantages, as will be elaborated on in Section 3, GNP does not guarantee that a host has a unique coordinate in a Cartesian coordinate system. Depending on the initial value used in the Simplex Downhill method, a single host may have different coordinates.

In this paper, we present a new Coordinate system called the *Internet Coordinate System (ICS)*. The distances from a host to beacon nodes are expressed as a distance vector, where the dimension of the distance vector is equal to the number of beacon nodes. As each beacon node defines an axis in the distance data space, the bases may be correlated. We apply the principal component analysis (PCA) to projects the distance data space into a new, uncorrelated and orthogonal Cartesian coordinate system of (much) smaller dimensions. The linear transformation essentially extracts topology information from delay measurements between beacon nodes and retains it in a new coordinate system. By taking the first several principal components (obtained in PCA) as the bases, we can construct the Cartesian coordinate system of smaller dimensions while retaining as much topology information as possible.

Based on the PCA-derived Cartesian coordinate system, we then propose a method to estimate the network distance between arbitrary hosts on the Internet. The network distances between beacon nodes are first analyzed to retrieve the principal components. The first several components are scaled by a factor (such that the Euclidean distances in the new coordinate system approximate the measured distances) and used as the new bases in the coordinate system. The coordinate of a host is then determined by multiplying its original distance vector to (a subset of) beacon nodes with the linear transformation matrix. As compared to GNP, ICS is computationally efficient because it only requires linear algebra operations. In addition, the location of a host is uniquely determined in the coordinate system. Another advantage of ICS is that it incurs smaller measurement overhead, as it does not require a host to make delay measurement to *all* the beacon nodes. Instead, a host may measure its distances only to a subset of beacon nodes. This is especially desirable in the case that some of the beacon nodes are not available (due to transient network partition and/or node failure). Finally, we show via Internet experimentation with real-life data sets that ICS is robust and accurate, regardless of the number of beacon nodes (as long as it exceeds certain threshold) and the complexity of network topology.

The rest of the paper is organized as follows. In Section 2, we provide the preliminary material and define a distance coordinate system using linear algebra. In Section 3, we give a summary of related work in the literature and motivate the need for a new Coordinate system. In Sections 4–5, we first introduce PCA and then propose the ICS architecture that enables construction of network topology in a coordinate system. Following that, we present in Section 6 experimental results, and conclude the paper in Section 7.

2. PRELIMINARY

The network topology can be modeled in a coordinate system based on the delay measured between hosts on the Internet. Each host measures the network distance (i.e., the round trip delay) to the other hosts using *ping* or *traceroute*. Under the assumption that there exist m hosts, a host $\mathcal{H}_i$ has a distance vector d_i as its coordinate in an m-dimensional system:

$$d_i = [d_{i1}, \ldots, d_{im}]^T \,, \tag{1}$$

where d_{ij} is the network distance measured by the i^{th} host to the j^{th} host and $d_{ii} = 0$. In general, $d_{ij} \neq d_{ji}$ because forward and reverse paths may have different characteristics. The overall system is represented by an m-by-m distance matrix D, whose i^{th} column is the coordinate of host $\mathcal{H}_i$:

$$D = [d_1, \ldots, d_m] \,. \tag{2}$$

Here D is a non-symmetric square matrix with zero diagonal entries. This representation is quite simple and intuitive, but contains too much redundant information as every host defines its own dimension in the coordinate system. In this paper, we will study how to represent network distances between hosts in a coordinate system of the least possible dimension, while retaining as much topological information as possible.

In a coordinate system, the generalized distance metric [6] is defined as

$$L_p(d_i, d_j) = \left(\sum_{k=1}^{m} |d_{ik} - d_{jk}|^p \right)^{\frac{1}{p}} . \tag{3}$$

Some of the most important metrics are the Manhattan distance L_1, the Euclidean distance L_2, and the Chebyshev distance L_∞. It has been shown that L_∞ can be expressed as

$$L_\infty(d_i, d_j) = \lim_{p \to \infty} L_p(d_i, d_j) = \max_k |d_{ik} - d_{jk}|.$$

3. RELATED WORK

3.1 Methods in the distance data space

Several methods have been proposed to estimate the distance between hosts on the Internet. These methods envision an infrastructure in which servers (beacon nodes) measure network distances between one another, and a client h_i (ordinary host) infers its distance to some other host h_j based on that distance information between servers. Hotz defined, for a host $\mathcal{A}$, a distance vector $d_a = [d_{a1}, \ldots, d_{am}]^T$ [3], where d_{ai} is the measured distance to the i^{th} beacon node for $i \in \{1, \ldots, m\}$ and m is the number of beacon nodes. Then, the network distance L between hosts $\mathcal{A}$ and $\mathcal{B}$ was shown to be bound by:

$$\max_i |d_{ai} - d_{bi}| \leq L \leq \min_i (d_{ai} + d_{bi}). \tag{4}$$

Note that the lower bound is the Chebyshev distance between the two vectors, d_a and d_b. Hotz also showed that the average of the upper and lower bounds generally gives a better estimation of the distance than each bound. Guyton *et al.* later applied Hotz's triangulation to distance calculation for locating nearby servers on the Internet [4].

A global architecture for estimating Internet host distances, called the Internet Distance Map Service *IDMaps*, was first proposed by Francis *et al.* [1]. The architecture separates beacon nodes (called *tracers*) that collect and distribute distance information from clients that use the distance map. Each tracer measures the distances to IP address prefixes (APs) that are close to itself. A client first determines its own AP and the autonomous system (AS) the AP is connected to. The client then runs a spanning-tree algorithm over the distance information gathered by tracers to find the shortest distance between its AS and the AS that the AP of the destination belongs to. This distance is taken as the estimated distance. Methods of this type (i.e., methods that represent network distances in a distance data space) neither analyze delay measurements nor infer network topology. Consequently, their performance depends heavily on the number and placement of beacon nodes. If the number of beacon nodes is small, the measurement performance may not be good.

In order to extract topological information, Ratnasamy *et al.* [7] proposed a binning scheme. A bin is defined as the list of beacon nodes in the order of increasing delay. The bin of a host indicates the relative distances to all the beacon nodes. For example, if the bin of a host is "$b_a b_c b_b$", beacon node b_a is the closest to the host, and b_b is the farthest to the host. The authors applied the binning scheme to the problems of constructing overlay networks and selecting servers. In the binning scheme, a host joins an overlay network node or selects a server whose bin is most similar to its own bin.

3.2 Methods that use the Cartesian coordinate system

Ng *et al.* proposed a Cartesian coordinate-based approach, called *Global Networking Positioning (GNP)* [2]. Instead of using the original network distances, GNP represents the location of each host in a N-dimensional Cartesian coordinate system, where N is the number of beacon nodes. The coordinate of a host is the distances from itself to the beacon nodes, and the distance between two hosts is calculated as the Euclidean distance in the Cartesian coordinate. The major advantage of representing network distances in a coordinate system is to extract topological information from the measured network distances. As a result, the accuracy in estimating the distance between two arbitrary hosts will be improved especially in the case that the number of beacon nodes is small.

Two optimization problems have been considered in GNP in order to obtain the coordinates of beacon nodes and hosts in the Cartesian coordinate system. The first problem obtains the coordinates of beacon nodes in GNP by minimizing the difference between the measured distance and the computed distance of any pair of beacon nodes in the Cartesian coordinate system:

$$J_1 = \sum_{i,j} \left(\tilde{d}_{ij} - L_2(d_i, d_j) \right)^2, \tag{5}$$

where $\tilde{d}_{ij}$ is the measured distance between the i^{th} and j^{th} beacon nodes, and d_i is the coordinate of the i^{th} beacon node in the Cartesian coordinate system. The second optimization problem determines the coordinate of an ordinary host $\mathcal{H}$ by minimizing the following cost function:

$$J_2 = \sum_{i} \left(\tilde{d}_{hi} - L_2(d_i, d_h) \right)^2, \tag{6}$$

where $\tilde{d}_{hi}$ is the measured distance between host $\mathcal{H}$ and the i^{th} beacon nodes, and d_h is the coordinate of the host $\mathcal{H}$. GNP tackles both optimization problems using the Simplex Downhill method [8]. Unfortunately, the Simplex Downhill method only gives a local minimum that is close to the starting value and does not guarantee that the result is unique in the case that the cost functions are not (strictly) convex. (The cost functions expressed in Eqs. (5) and (6) are not strictly convex.) It is stated in [2] that the first optimization problem may have an infinite number of solutions, and any solution is sufficient. This implies that the Simplex Downhill method is used to find one of the local minima. If the solution to the first optimization problem is a good approximation of a global minimum, the coordinates of beacon nodes thus calculated suffice in the first problem. A host in GNP may have different coordinates depending on the starting values used in the Simplex Downhill method. However, this is not the case in the second optimization problem. The fact that ordinary hosts may have non-unique coordinates may lead to estimation inaccuracy. We demonstrate the problem in the following example.

EXAMPLE 1. Problem with GNP: *Consider four hosts, two of which are located in one autonomous system (AS), and the other two in another AS. Also assume (for demonstration purpose) that the distance between two hosts in the same AS is 1 while the distance between two hosts in different ASs is 3. Then the topology can be expressed using the following distance matrix D:*

$$D = \begin{bmatrix} 0 & 1 & 3 & 3 \\ 1 & 0 & 3 & 3 \\ 3 & 3 & 0 & 1 \\ 3 & 3 & 1 & 0 \end{bmatrix}.$$

The first cost function J_1 in two-dimensional coordinate system is

$$\begin{aligned} J_1 &= \sum_{(i,j)=(1,2),(3,4)} \left(1 - \sqrt{\sum_{k=1}^{2} (d_{ik} - d_{jk})^2} \right)^2 \\ &+ \sum_{(i,j)=(1,3),(1,4),(2,3),(2,4)} \left(3 - \sqrt{\sum_{k=1}^{2} (d_{ik} - d_{jk})^2} \right)^2. \end{aligned}$$

We solve the optimization problem using the 'fminsearch' function in Matlab, which implements the Simplex Downhill method, with the starting values, $d_1^0 = [0,0]^T$, $d_2^0 = [1,1]^T$, $d_3^0 = [-1,-1]^T$, and $d_4^0 = [0,0]^T$. The coordinates

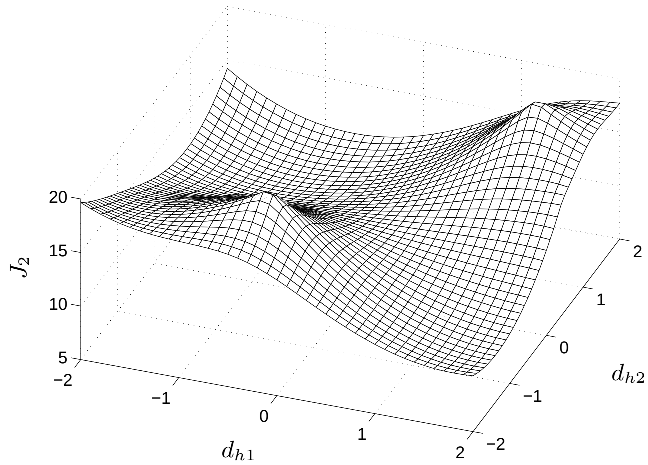


Figure 1: The cost function for the coordinate of an ordinary host in Example 1

.

of the beacon nodes calculated with this set of starting values are $d_1 = [0.4433, 2.0048]^T$*,* $d_2 = [1.2262, 1.4248]^T$*,* $d_3 = [-0.5137, -0.9240]^T$*, and* $d_4 = [-1.2966, -0.3440]^T$*. Note that* $L_2(d_1, d_2) = 0.9743 \approx 1$*,* $L_2(d_1, d_3) = 3.0812 \approx 3$ *and so on.*

Now assume that a host $\mathcal{H}$ *measures its distances to four beacon nodes, and obtains a distance vector* $\tilde{d}_h = [1, 4, 1, 4]^T$*. The second cost function* J_2 *in the second optimization problem becomes*

$$J_2 = \sum_{i=1,3}\left(1 - \sqrt{\sum_{k=1}^{2}(d_{ik} - d_{hk})^2}\right)^2 + \sum_{i=2,4}\left(4 - \sqrt{\sum_{k=1}^{2}(d_{ik} - d_{hk})^2}\right)^2 .$$

Figure 1 depicts the cost function J_2 *with respect to* d_{h1} *and* d_{h2}*. The cost function has two local minima at (1.2866, -0.9130) and (-1.3571, 1.9938). Therefore,* d_h *can be either* $[1.2866, -0.9130]^T$ *or* $[-1.3571, 1.9938]^T$ *depending on the starting values of the Simplex Downhill method. If the starting value is (1,-1), the Simplex Downhill method renders the former local minimum (1.2866, -0.9130). This implies that GNP does not guarantee a unique mapping from the original distance vector to the Cartesian coordinate.*

Our proposed approach, ICS, shares the similarity with GNP in that it also represents locations of hosts in the Cartesian coordinate system instead of a distance data space, and consequently, can extract topological information from measured network distances. ICS, however, provides a unique mapping from the distance data space to the Cartesian coordinate system (and thus yields a more accurate representation). In addition, it has the following advantages:

- With the use of principal component analysis (PCA), a host can calculate its coordinates by means of basic linear algebra such as the singular value decomposition and matrix multiplication. The computational overhead is reduced.
- Unlike all the other previous work, a host does not have to measure its distance to *all* the beacon nodes, but can determine its coordinate by measuring the distances to a subset of beacon nodes. The message overhead is reduced.

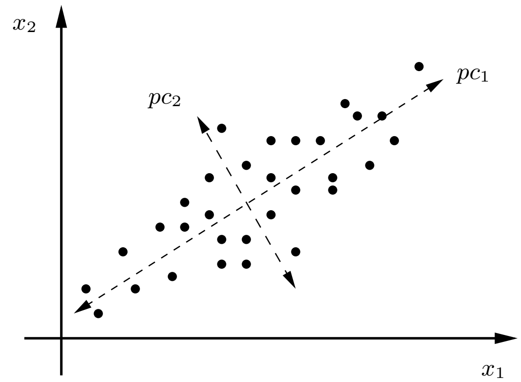


Figure 2: Example of the principal component analysis

It has come to our attention that Tang *et al.* also applied principal component analysis to project distance measurements into a Cartesian coordinate system with smaller dimensions [5]. The authors considered the coordinate of a host in the coordinate system as the distances to *virtual landmarks* while the coordinate in the distance data space represents the distances to actual beacon nodes (landmarks). However, unlike GNP and ICS, the Euclidean distance between two hosts in this scheme does not approximate the real round trip time but reflects the relative proximity. For the sake of scalability, the authors also devised a coordinate exchanging method among multiple coordinate systems.

4. PRINCIPAL COMPONENT ANALYSIS (PCA)

We analyze the distance matrix D in Eq. (2) to extract topological information in a coordinate system. In the previous example of four hosts, the dimension of the distance matrix D is four. As hosts in the same AS are very close to each other, the distance can be represented in a two-dimensional space by projecting their coordinates into two-dimensional space. The dimensionality depends not on the dimension m of the distance matrix D but on the network topology, and can be much smaller than m.

We apply principal component analysis (PCA) [9, 11, 12] to reduce the dimension of the distance matrix while retaining as much topological information as possible. In a nutshell, PCA transforms a data set that consists of a large number of (possibly) correlated variables to a new set of uncorrelated variables, *principal components*, which can characterize the network topology. The principal components are ordered so that the first several components have the most important features of the original variables. The k^{th} principal component is interpreted as the direction of maximizing the variation of projections of measured distance data while orthogonal to the first $(k-1)^{th}$ principal components [11].

EXAMPLE 2. *Figure 2 gives an example of performing PCA for two correlated variables,* x_1 *and* x_2*. With the use of*

PCA, we obtain two principal components, pc_1 and pc_2. As shown in Fig. 2, the first principal component pc_1 represents the direction of the maximum variance. The one-dimensional linear representation is calculated by projecting the original data onto pc_1.

These principal components can be obtained by singular value decomposition (SVD). The singular value decomposition of D in Eq. (2) is obtained by

$$D = U \cdot W \cdot V^T, \quad (7)$$
$$W = \begin{bmatrix} \sigma_1 & & & \\ & \sigma_2 & & \\ & & \ddots & \\ & & & \sigma_m \end{bmatrix},$$

where U and V are column and row orthogonal matrices, and σ_i's are the singular values of D in the decreasing order (i.e., $\sigma_i \geq \sigma_j$ if $i < j$). Note that $D^T D = (UWV^T)^T(UWV^T) = V(W^T W)V^T$. This means that the eigenvectors of $D^T D$ make up V with the associated (real nonnegative) eigenvalues of the diagonal of $W^T W$ [10]. Similarly, $DD^T = U^T(WW^T)U$. The columns of the $m \times m$ matrix $U = [\ u_1, \ldots, u_m\ ]$ are the principal components and become the orthogonal basis of the new subspace. By using the first n columns of U denoted by U_n, we project the m-dimensional space into a new n-dimensional space:

$$c_i = U_n^T \cdot d_i = [\ u_1,\ \ldots,\ u_n\ ]^T \cdot d_i. \quad (8)$$

Example 3. *Consider the four hosts with the following distance matrix D.*

$$D = \begin{bmatrix} 0 & 1 & 3 & 3 \\ 1 & 0 & 3 & 3 \\ 3 & 3 & 0 & 1 \\ 3 & 3 & 1 & 0 \end{bmatrix}$$

We can obtain the singular value decomposition.

$$U = \begin{bmatrix} -\frac{1}{2} & -\frac{1}{2} & \frac{1}{\sqrt{2}} & 0 \\ -\frac{1}{2} & -\frac{1}{2} & -\frac{1}{\sqrt{2}} & 0 \\ -\frac{1}{2} & \frac{1}{2} & 0 & -\frac{1}{\sqrt{2}} \\ -\frac{1}{2} & \frac{1}{2} & 0 & \frac{1}{\sqrt{2}} \end{bmatrix},\ W = \begin{bmatrix} 9 & 0 & 0 & 0 \\ 0 & 7 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}.$$

The original distance vector of the first host is $d_1 = [0, 1, 3, 3]^T$. With the use of Eq. (8), we can calculate the coordinate of the first host in a two-dimensional coordinate system as

$$c_1 = U_2^T d_1 = \begin{bmatrix} -\frac{1}{2} & -\frac{1}{2} & -\frac{1}{2} & -\frac{1}{2} \\ -\frac{1}{2} & -\frac{1}{2} & \frac{1}{2} & \frac{1}{2} \end{bmatrix} \cdot \begin{bmatrix} 0 \\ 1 \\ 3 \\ 3 \end{bmatrix} = \begin{bmatrix} -\frac{7}{2} \\ \frac{5}{2} \end{bmatrix}.$$

Similarly $c_1 = c_2 = [-\frac{7}{2}, \frac{5}{2}]^T$ and $c_3 = c_4 = [-\frac{7}{2}, -\frac{5}{2}\ ]^T$. Note that PCA assigns the same coordinate to the two hosts in the same AS. When $n = 4$, $U_4 = U$, $c_1 = [-7, 5, -\sqrt{2}, 0]$, $c_2 = [-7, 5, \sqrt{2}, 0]$, $c_3 = [-7, -5, 0, \sqrt{2}]$, and $c_4 = [-7, -5, 0, -\sqrt{2}]$. In this case (m=n), the mapping $c_i = U^T \cdot d_i$ is isometric (i.e., $L_2(d_i, d_j) = L_2(c_i, c_j)$), and thus the two spaces spanned by d_i's and c_i's are the same from the perspective of geometry.

4.1 Dimensionality

One important issue that should be addressed in representing network distances in a n-dimensional coordinate system is how to determine the adequate degree, n, of dimensions in the coordinate system. Generally this problem of determining the number of principal components has not been extensively studied, and is usually application-dependent [13]. One of the commonly adopted criteria is the cumulative percentage of variation that the selected principal components contribute [9]. The percentage, t_k, of variation accounted for by the first k principal components is defined by

$$t_k = 100 \times \frac{\sum_{j=1}^{k} \sigma_j}{\sum_{j=1}^{m} \sigma_j}. \quad (9)$$

Usually a cut-off value, t^* of cumulative percentage of variation is pre-determined, and n is determined to be the smallest integer such that $t_n \geq t^*$. In the previous example, $t_1 = 50\%$, $t_2 = 89\%$, $t_3 = 94$ %, and $t_4 = 100$ %. If t^* is set to 80 %, then we have $n = 2$.

Table 1: Average proximity in original geometry space D

Metric	NPD (m = 33)	NLANR (m = 113)
L_1	5.818	6.964
L_2	6.545	6.495
L_∞	12.151	5.504

4.2 Experimental Results

To investigate whether or not PCA can be used to represent the network distances on the Internet in a coordinate system of smaller dimensions and yet still retain as much topological information as possible, we apply PCA to two real-life data sets:

- NPD-Routes-2 data set [14]: contains Internet route measurements obtained by *traceroute*. The measurements were made between 33 Internet hosts in the Network Probe Daemon (NPD) framework from November 3, 1995, to December 21, 1995. We obtain the distance matrix D in Eq. (2) by taking (for each pair of hosts) the minimum value of measured round trip times (RTTs) in order to filter out queuing delay.

- NLANR: contains the RTT, packet loss, topology, and on-demand throughput measurements made under the Active Measurement Project (AMP) at National Laboratory for Applied Network Research (NLANR). More than 100 AMP monitors are used to make the measurements [15]. The round trip times between all the monitors are measured every minute, and are processed once a day. We use one of the NLANR RTT data sets measured between 113 AMP monitors on April 9, 2003.

We first compare different distance metrics with respect to their quality of representing topological information. We use three distance metrics, L_1, L_2, and L_∞ in Eq. (3). We calculate for each host the distances L_1, L_2, and L_∞ to all the other hosts, and determine its closest host based on the distance calculated in the coordinate system. As the "closest" host calculated under the various distance metrics may not be the actual closest host, we define the notion of proximity to measure the quality of representing topological information. If the host calculated to be the closest is the

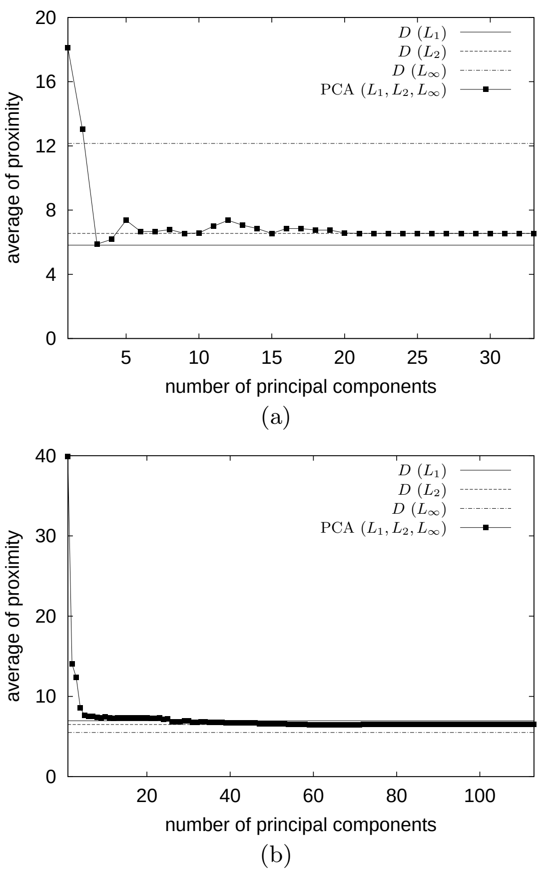


Figure 3: Average proximity for the NPD data set ((a)) and the NLANR data set ((b)) under different distance metrics.

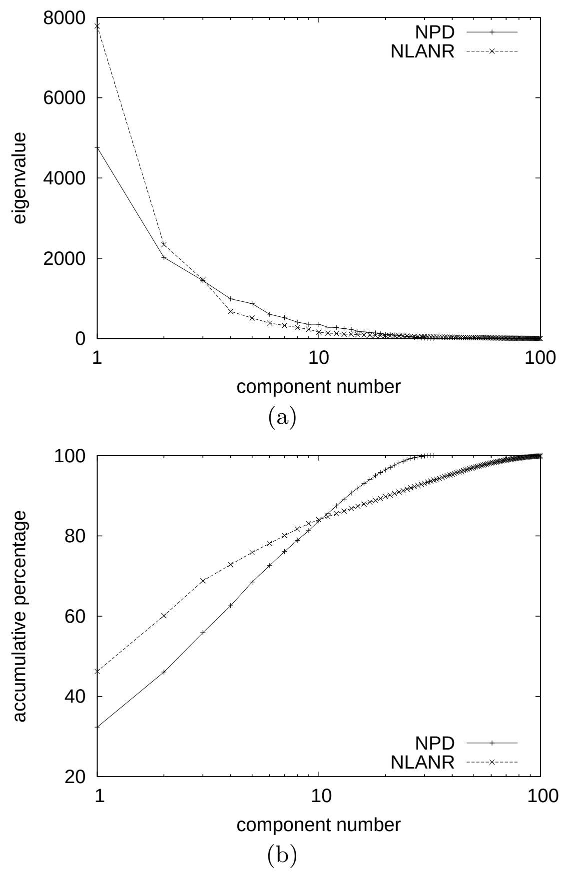


Figure 4: Eigenvalues and cumulative percentage of variation for the NPD data set and the NLANR data set.

closest, the proximity is set to 1. Similarly, if the host calculated to be the closest turns out to be the k^{th} closest, the proximity is k. We average, for each distance metric used, the proximity over all the hosts.

Table 1 gives the average proximity in the original distance space, whose dimension is $m = 33$ and 113 for the NPD and NLANR data sets, respectively. In the NPD data set, L_1 gives the best performance — the host calculated to be the closest is the 5.818^{th} closest host averagely. In the NLANR data set, L_∞ gives the best performance. These results show that the accuracy of representing topological information in a distance data space depends heavily on the distance metric.

Next we study the (in)effectiveness of using PCA to represent network distances. Figure 3 gives the average proximity with respect to the number of principal components for the NPD and NLANR data sets. As shown in Fig. 3 (a), when the number of principal components is greater than 3, the proximity is almost the same as that in the original distance data space. This means that the topological information is effectively represented in a 3-dimensional space instead of in a 33-dimensional space. Another important observation is that the average proximity in the new coordinate system of smaller dimensions remains the same regardless of the distance metric used. The reason that the proximity is independent of the distance metric is due to the fact that PCA finds a set of uncorrelated bases to represent the topological information. A similar trend can be observed in Fig. 3 (b) in which the proximity is almost the same as that in the original distance data space when the number of principal components is larger than 10.

Figure 4 plots the eigenvalues and their corresponding cumulative percentage of variation. The largest eigenvalues are 4760.0 and 7787.3, respectively, for the NPD and NLANR data sets. If we set a cut-off threshold of $t^* = 80$ %, the smallest value of n that achieves the threshold for each data set is, respectively, 9 and 7. In this case, $\sigma_9 = 354.7$, and the average proximity is 6.54 for the NPD data set, and $\sigma_7 = 325.2$ and the average proximity is 7.49 for the NLANR data set.

In summary, we show in this section that the Internet distance can be modeled, with the use of PCA, in a Cartesian space that uses a (smaller) set of uncorrelated bases. Moreover, we show that the new coordinate system is less susceptible to the distance metrics used in representing topological information.

5. INTERNET COORDINATE SYSTEM

5.1 Overview

We first present a basic architecture for the Internet coordinate system (ICS). As mentioned in Section 1, the objectives of ICS are i) to infer the network topology based on delay measurement and ii) to estimate the distance between hosts without direct measurement. Succinctly, the architecture for ICS consists of a number of beacon nodes, that collect and analyze distance information. Figure 5 gives an example architecture of ICS with five beacon nodes. Beacon nodes periodically measure round trip times (RTTs) to other beacon nodes and construct a coordinate system. The coordinates of beacon nodes are then calculated, with the use of PCA, based on the measured RTT data among five beacon nodes. We will elaborate on how to calculate the coordinates of beacon nodes in Section 5.2.

An ordinary host determines its own location in ICS by measuring its delays to either the entire or partial set of beacon nodes and obtains a distance vector. As exemplified in Fig. 5, host 1 measures its distance to five beacon nodes, and obtains a five-dimensional distance vector. The location of the host in ICS is then calculated by multiplying the distance vector with a transformation matrix. (We will elaborate on how the transformation matrix is derived and distributed in Section 5.3.) After calculating its own coordinate, host 1 may report its coordinate to a DNS-like server that keeps coordinates of ordinary hosts. To estimate the network distance to some other host without direct measurement, host 1 may query this DNS-like server which can then easily determine the estimated distance as long the coordinate of the other host is kept. In the same manner, host 1 can also infer which host is closer to itself.

5.2 Calculating the Coordinates of Beacon Nodes

We construct the Internet Coordinate system based on the measured network distances between m beacon nodes. PCA presented in Section 4 is applied to reduce the distance data space to a new coordinate system of (much) smaller dimensions.

Each beacon node measures its distances to the other beacon nodes, and obtains a m-dimensional distance vector d_i in Eq. (1), of which the j^{th} element d_{ij} is the measured distance to the j^{th} beacon node. An administrative node, which can be elected among beacon nodes, aggregates the distance vectors of all the beacon nodes, and obtains the distance matrix D in Eq. (2). Then, the distance matrix is decomposed into three matrices U, W, and V in Eq. (7). Using the first n principal components, the coordinate of a beacon node is calculated as $c_i = U_n d_i$ in Eq. (8). As shown in Section 4.2, this coordinate preserves topological information.

As the distance between two beacon nodes calculated in the coordinate system may not be the same as the actual measured distances. For instance, $L_2(c_1, c_3) = 5 \neq 3$ when $n = 2$ in Example 1. To use the coordinates for distance estimation, we apply a simple linear operation, $\bar{c}_i = \alpha c_i + \beta$ so as to minimize the discrepancy between the distance represented in the coordinate system and the measured distance. As a translation operation does not affect the distance between two coordinates, we only consider the scaling operation with a scaling factor α, i.e., $\beta = 0$. The optimal scaling factor $\alpha^*(n)$ that minimizes the discrepancy between the Euclidean distance in the new coordinate system of dimension n and the measured delay, i.e., $L_2(\bar{c}_i, \bar{c}_j) \approx d_{ij}$ for all i and $j \in \{1, \ldots, m\}$, can be determined by minimizing the following objective function $J(\alpha)$:

$$J(\alpha) = \sum_i^m \sum_j^m (L_2(\alpha c_i, \alpha c_j) - d_{ij})^2 \quad (10)$$

whose positive solution, α^*, is simply

$$\alpha^*(n) = \frac{\sum_i^m \sum_j^m d_{ij} L_2(c_i, c_j)}{\sum_i^m \sum_j^m L_2(c_i, c_j)^2}. \quad (11)$$

The transformation matrix $\bar{U}_n$ from a distance vector in the distance data space to the coordinate in ICS is then defined as

$$\bar{U}_n = \alpha^*(n) U_n = \frac{\sum_i^m \sum_j^m d_{ij} l_{ij}}{\sum_i^m \sum_j^m l_{ij}^2} U_n, \quad (12)$$

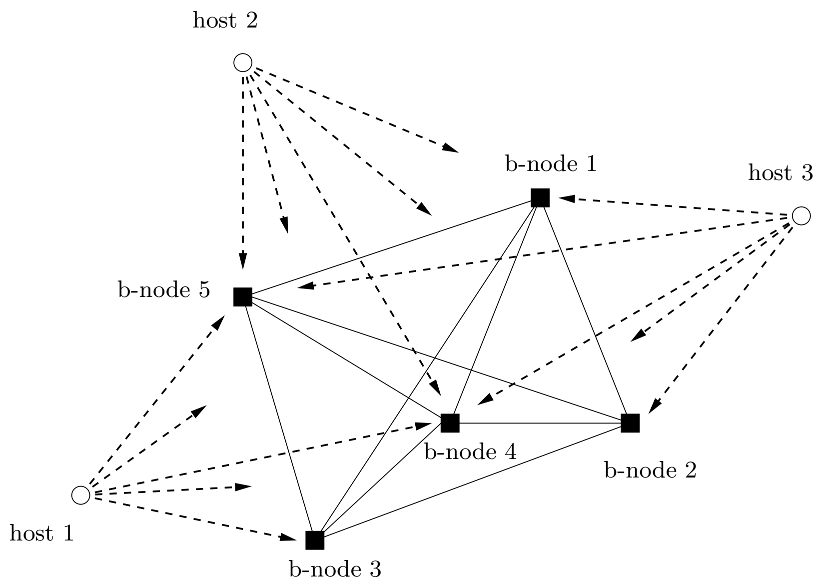


Figure 5: An example architecture for the proposed Internet coordinate system (five beacon nodes and three ordinary hosts).

where $l_{ij} = L_2(U_n^T d_i, U_n^T d_j)$ and $U_n = [u_1, \ldots, u_n]$. The transformation matrix is obtained from the distance matrix D between beacon nodes and its singular value decomposition. The coordinates of beacon nodes are then calculated as $\bar{c}_i = \bar{U}_n^T d_i$ for all $i \in \{1, \ldots, m\}$.

In summary, the procedure taken to calculate the coordinates of beacon nodes is as follows:

(S1) Every beacon node measures the round trip times to the other beacon nodes periodically.

(S2) An administrative node aggregates the delay information and obtains the distance matrix in Eq. (2).

(S3) The administrative node applies PCA in Eq. (7) to the distance matrix.

(S4) The administrative node determines the dimension of the coordinate system using the cumulative percentage of variation defined in Eq. (9) (with a pre-determined threshold value).

(S5) The administrative node calculates the transformation matrix in Eq. (12) from Eq. (8) and Eq. (11).

Note that the administrative node may be replicated (perhaps in a hierarchical manner) to enhance fault tolerance and availability. This subject is outside the scope of this paper, but is warrant of further investigation.

Example 4. *Assume that the four hosts in Example 1 are beacon nodes. When $n = 2$, $c_1 = c_2 = [-3.5, 2.5]$ and $c_3 = c_4 = [-3.5, -2.5]^T$. By Eq. (11), the scaling factor α is 0.6, and the transformation matrix $\bar{U}_2$ is*

$$\bar{U}_2 = \begin{bmatrix} -0.3 & -0.3 & -0.3 & -0.3 \\ -0.3 & -0.3 & 0.3 & 0.3 \end{bmatrix}^T .$$

Therefore, $\bar{c}_1 = \bar{c}_2 = [-2.1, 1.5]$ and $\bar{c}_3 = \bar{c}_4 = [-2.1, -1.5]$. The distances between two hosts in different ASs is exactly 3. When n=4, α = 0.5927, $L_2(\bar{c}_1, \bar{c}_2) = L_2(\bar{c}_3, \bar{c}_4) = 0.8383$, and $L_2(\bar{c}_1, \bar{c}_3) = L_2(\bar{c}_1, \bar{c}_4) = L_2(\bar{c}_2, \bar{c}_3) = L_2(\bar{c}_2, \bar{c}_4) = 3.0224$.

5.3 Determining The Coordinate of A Host

The procedure that a host takes to determine its coordinate in ICS is as follows: A host

(H1) Obtains the list of beacon nodes and the transformation matrix (Eq. (12)) from the administrative node.

(H2) Measures the round trip times to all the beacon nodes using *ping* or *traceroute*. (We will discuss how to reduce the number of measurements in Section 5.4.)

(H3) Calculates the coordinate by multiplying the measured distance vector with the transformation matrix.

In (H2), a host $\mathcal{A}$ obtains an m-dimensional distance vector

$$l_a = [l_{a1}, \ldots, l_{am}]^T , \tag{13}$$

where l_{ai} denotes the delay measured between host $\mathcal{A}$ and the i^{th} beacon node. Then in (H3) the coordinate, x_a, of host $\mathcal{A}$ is calculated with the transformation matrix $\bar{U}_n$ in Eq. (12) as

$$x_a = \bar{U}_n^T \cdot l_a. \tag{14}$$

Example 5. *A host $\mathcal{A}$ measures its round trip times to the four beacon nodes in Example 4. Assume that host $\mathcal{A}$ is closer to the AS where the first two beacon nodes reside, and obtains a distance vector of $l_a = [1, 1, 4, 4]^T$. By Eq. (14), $x_a = [-3, 1.8]^T$. In the case of $n = 2$, the estimated distances between host $\mathcal{A}$ and beacon nodes are $L_2(\bar{c}_1, x_a) = L_2(\bar{c}_2, x_a) = 0.94$ and $L_2(\bar{c}_3, x_a) = L_2(\bar{c}_4, x_a) = 3.42$. On the other hand, assume that host $\mathcal{B}$ is far from all four beacon nodes, and obtains a distance vector of $l_b = [10, 10, 10, 10]^T$. In this case, $x_b = [-12, 0]^T$, and $L_2(\bar{c}_i, x_b) = 10.01$ for $i = 1, \ldots, 4$.*

5.4 Reducing The Number of Measurements

To discover accurately the topology of the Internet, a sufficient number of beacon nodes should be judiciously placed on the Internet. (Note that PCA is able to extract essential topological information from a set of (perhaps correlated) delay measurements. However, it does not preclude the important task of placing beacon nodes properly on the Internet so as to represent the network topology accurately.) On the other hand, for scalability reason, it is not desirable that a client has to measure its round trip times to *all* the beacon nodes. To reduce the measurement overhead incurred by a host, it would be desirable that a host measures the distance from itself to a subset of beacon nodes. This also allows ICS to operate even in the case that some of the beacon nodes are not available (due to transient network partition and/or node failure).

By Eq. (14), the transformation matrix (Eq. (12)) and the original distance vector (Eq. (13)) are needed to calculate the coordinate of a host. The transformation matrix is fixed in ICS once it is calculated by the administrative node. If host $\mathcal{A}$ makes delay measurements only to a subset, $\mathcal{N}$, of beacon nodes, the missing elements in l_a, i.e., l_{ai}, $i \notin \mathcal{N}$, have to be inferred. We present the following two methods:

(M1) Host $\mathcal{A}$ randomly chooses k beacon nodes ($k < m$) and measures its distances to this subset, $\mathcal{N}$, of beacon nodes. (In our experiments, we will investigate the effect of k on the estimation performance.) Instead of calculating the coordinate by itself, host $\mathcal{A}$ then transmits the distance vector l_a with $m - k$ missing elements to the administrative node. For each missing element l_{ai} in l_a, the administrative (i) selects, among all the beacon nodes that are in $\mathcal{N}$, a beacon node (say the j^{th} beacon node) that is closest to the i^{th} beacon node, (ii) replaces the missing element l_{ai} with l_{aj}, and (iii) calculates the coordinate on the behalf of host $\mathcal{A}$.

(M2) The administrative node can specify, for host $\mathcal{A}$, a list of specific beacon nodes to which host $\mathcal{A}$ should make delay measurements. The rationale is that if two beacon nodes are close enough, host $\mathcal{A}$ does not have to measures its distances to both beacon nodes. In order to find k groups each of which consists of beacon nodes that are close to one another, the administrative node applies a hierarchical clustering technique [16] to the distance matrix, and selects the median beacon node for each cluster. The administrative node then sends host $\mathcal{A}$ a list of k median beacon nodes and a list of k clusters, with the i^{th} median beacon node corresponding to the i^{th} cluster. Host $\mathcal{A}$ measures its distances only to the k median beacon nodes, and uses the distance to the i^{th} median beacon node as that to the other beacon nodes in the same cluster.

The performance of the above two partial measurement methods depends heavily on how well the missing elements in l_a are represented. In order to improve the performance, instead of directly using the network distance measured to the closest/median beacon node, we can use Hotz's triangulation method (Section 3). As a beacon node $\mathcal{B}$ that is not in $\mathcal{N}$ has already measured its distances to other beacon nodes, the distance between the host and node $\mathcal{B}$ can be estimated by Hotz's triangulation method.

6. EMPIRICAL STUDY

To validate the effectiveness of ICS in inferring the Internet topology, we conduct experiments using both an empirical data set (NLANR) [14] and a synthetic data set (GT-ITM) [17]. As discussed in Section 4.2, the NLANR data set contains real delay data measured by *ping*. The GT-ITM data set, on the other hand, is obtained by the GT-ITM topology generator [17] and the *ns*-2 simulator [18]. The quality of a coordinate system can be affected by several factors such as the number and distribution of beacon nodes and the complexity of the network topology. With the use of the GT-ITM topology generator, we are able to study ICS under a wide variety of network topologies, and investigate the effect of network topology on the performance of ICS.

For each data set, we randomly select m beacon nodes ($3 \leq m \leq 30$). If beacon nodes are well distributed or selected with respect to certain clustering criterion, the performance is expected to be better, as was done in [2]. However, as beacon nodes may not be practically placed at any desirable location on the Internet, we believe it is more reasonable to include all the beacon nodes available on the Internet, but provide a method to enable a host to only measure its distances to a subset of beacon nodes. As such, we assume that beacon nodes are randomly selected in these experiments.

We compare ICS against with IDMaps, Hotz's triangulation, and GNP with respect to the estimation error $\mathcal{E}$ defined as

$$\mathcal{E} = \sum_{i,j \in \{1,\ldots,H\}} \frac{|\tilde{d}_{ij} - L(i,j)|}{\tilde{d}_{ij}},$$

where H is the number of hosts in the data sets, $\tilde{d}_{ij}$ is the measured distance, and $L(i,j)$ is the estimated distance between the i^{th} and j^{th} hosts. IDMaps, Hotz's triangulation, GNP, and ICS are implemented as follows:

- IDMaps: Suppose hosts $\mathcal{A}$ and $\mathcal{B}$ are close to the i^{th} and the j^{th} beacon nodes (called *tracers* in IDMaps), respectively. The corresponding distances are denoted as d_{ai} and d_{bj}. Then the estimated distance is $d_{ai} + d_{bj} + d_{ij}$, where d_{ij} is the distance between the i^{th} and j^{th} beacon nodes.
- Hotz's triangulation: With Eq. (4), we calculate three Hotz's distances, i.e., the lower bound (denoted as lb), the upper bound (denoted as ub), and the average of the two bounds (denoted as avg).
- GNP: We solve the two optimization problems minimizing J_1 in Eq. (5) and J_2 in Eq. (6) using the 'fminsearch' function in Matlab (which implements the Simplex Downhill method). We vary the dimension of the coordinate system from $n = 2$ to 10, and report only the most representative results.
- ICS: We evaluate both the full and partial measurement methods using the same range of the coordinate dimension as in GNP. In the partial measurement method, the number of measurements by a host is set to be $k = n, 2n$, and $3n$, where n is the dimension of the coordinate system. Method (*M1*) is used as the partial measurement method, and the missing elements in the distance vector l_a of host $\mathcal{A}$ are estimated by Hotz's triangulation (as was discussed in the last paragraph of Section 5).

6.1 Results for the NLANR data

Figure 6 (a) gives the estimation errors of IDMaps and Hotz's triangulation. The error obtained by IDMaps is quite large, but gradually decreases from 1.32 at $m = 3$ to 0.40 at $m = 30$. As the estimate is calculated by the sum of the three distances $d_{ai} + d_{bj} + d_{ij}$, if the two beacon nodes are on the shortest path, the estimate well approximates the network distance. This accounts for the fact that the estimate becomes more accurate as m increases. The upper bound of Hotz's triangulation exhibits the same trend as IDMaps. As m increases, the probability that the beacon nodes are on the shortest path between two hosts also increases. The lower bound is quite accurate when m is small. However, the estimation error increases as m increases. Consistent with the findings in [4], the average of the two bounds renders a more accurate estimate of the network distance, and is less susceptible to the number of beacon nodes.

Figure 6 (b) gives the estimation errors of GNP, ICS with the full measurement method, and Hotz's triangulation with the average of the two bounds. GNP performs better than Hotz's triangulation when the number of beacon nodes is small ($m \leq 15$). However, its estimation error increases as m increases, and becomes almost the same as that of Hotz's triangulation. This is probably due to the fact that a local minimum (rather than a global minimum) is selected in the optimization problems. Consider, for example, the case that there exist twenty beacon nodes and the dimension of the coordinate system is five. The cost function J_1 is minimized in a hundred-dimensional vector space, i.e., the number of variables in the coordinates of beacon nodes is 100. In general, an optimization problem of high dimensions easily converges to a local minimum, which in turn leads to inaccuracy in the coordinates of hosts, as explained in Section 3.2. ICS gives the best performance. In most cases, it incurs lower estimation errors than IDMaps. It gives the same performance as GNP when $m < 15$ and better performance when $m \geq 15$. Here, we select the dimension of the coordinate system to be five as the improvement is marginal when $n \geq 5$ as shown in the next figures.

Figure 7 depicts the effect of the dimension of the coordinate system on the performance of ICS ((a)) and GNP ((b)). The estimation error of ICS is the largest when the dimension of the coordinate system is two ($n = 2$), improves as the network topology is represented in higher dimensional space ($n > 2$), but the improvement levels off when $n \geq 6$. The estimation error of GNP is the smallest when $n = 4$, and is even slightly better than that of ICS in the range of $5 \leq m \leq 16$. Note also that the estimation error of GNP when $n = 6$ is much larger than that when $n = 4$. This is again due to the reason that the number of variables increases as n increases, and shows that the accuracy of GNP depends on the selection of the dimension of the coordinate system (i.e., the number of beacon nodes).

Figure 8 gives the results of ICS with the use of partial measurement method. The number of measurements made by a host is now proportional to the coordinate dimension, i.e. $k = \min(n, m)$ in (a) and $k = \min(2n, m)$ in (b). As shown in Fig. 8 (a), when $n = 6$, a client measures its distances to six beacon nodes regardless of m, and the average of the estimation errors is increased by 30.2 % (from 0.34489 in Fig. 7 (a) to 0.46692). When the number of measurements is increased twice in Fig. 8 (b), the average of the estimation errors is increased by 19.7 % in the case of $n = 6$. An interesting result is that the estimation error does not become larger even when m is large (e.g, $m > 15$). This is because the coordinate system obtained with the use of more beacon nodes is more accurate.

6.2 Results for the GT-ITM data

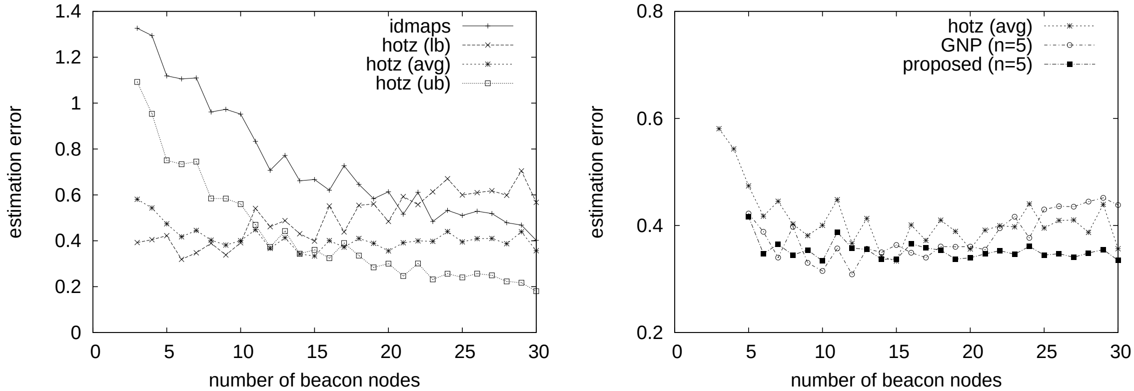
We now investigate the effect of the level of topology complexity on the distance estimation. As mentioned in [17], the GT-ITM topology generator can be used to create three types of graphs: flat random graphs, hierarchical graphs, and transit-stub graphs. We generate two-level and three-level hierarchical graphs with 400 nodes. Note that each graph has the same number of nodes, but three-level hierarchical graphs represent more complex network topologies.

Figure 9 (a) and (b) depict the performance of IDMaps, Hotz's triangulation, GNP, and ICS under the two-level hierarchical topology. As shown in Fig. 9 (a), methods that represent network topology in a distance data space give large estimation errors when the number of beacon nodes m is small, and their performance gradually improves as m increases. Among IDMaps and the three versions of Hotz's triangulation, the lower bound of Hotz's triangulation gives the best performance. As shown in Figure 9 (b), between the two coordinate-system-based approaches, GNP renders large estimation errors, and the errors increase as m increases. The estimation error of ICS, on the other hand, is 0.30 at $m = 5$, decreases as m increases, and becomes 0.17 at $m = 30$.

As shown in Fig. 9 (c) and (d), all the approaches, except ICS, give larger estimation errors under three-level hierarchical topologies. In particular, the performance of GNP becomes worse. ICS gives almost the same performance as in two-level hierarchical topologies. This result shows that PCA which ICS is built upon can effectively extract topological information than the minimization optimization of cost functions J_1 and J_2 in Eq. (5) and Eq. (6) used in GNP.

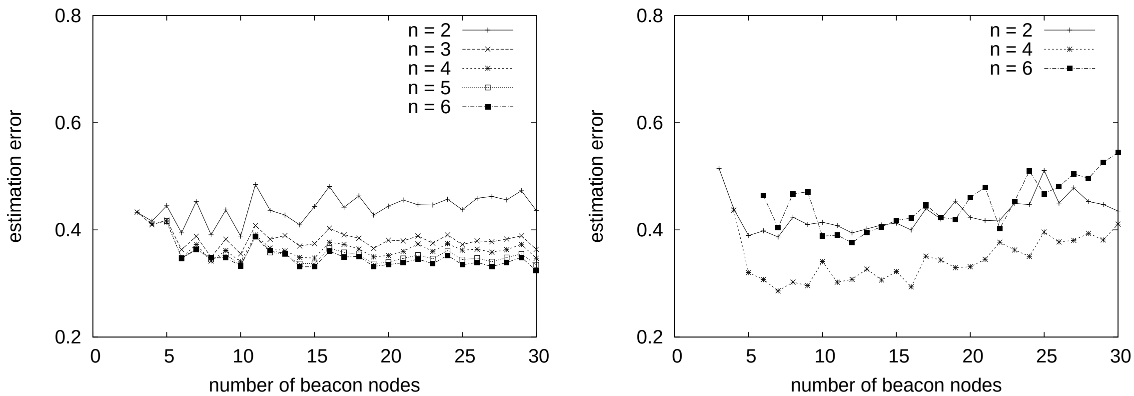
Figure 10 depicts the effect of the coordinate dimension on the performance of ICS with the full and partial measurement methods. The number of measurements made in the partial measurement method is set to $k = 2n$. Under all the cases, as the coordinate dimension n increases, the estimation errors decrease. As shown in Fig. 10 (a), there is virtually no performance improvement when $n \geq 3$, which implies that a three-dimensional coordinate space is sufficient to represent the two-level hierarchical topology. However, when the partial measurement method is applied, the estimation error increases from 0.209 to 0.407 in the case of $n = 3$ (94.7%). This means that even though a three-dimensional space is sufficient to represent the network topology, the number of measurements required should be larger than six in order to determine the coordinates of hosts accurately. When the number of measurements is twelve ($k = 12$) in the six-dimensional space, the estimation errors increase by 32.9% (from 0.188 to 0.250). As shown in Fig. 10 (c), the estimate made by ICS is quite accurate under the three-level hierarchical topology, and the errors decrease as n increases. As expected as shown in Fig. 10 (d), the estimation errors are larger when partial measurement is used.

In summary, IDMaps and the upper bound of Hotz's triangulation are inaccurate for small values of m, but yield better performance as m increases. In contrast, the lower bound of Hotz's triangulation is accurate for small values of m for the NLANR and GT-ITM data sets, and the errors



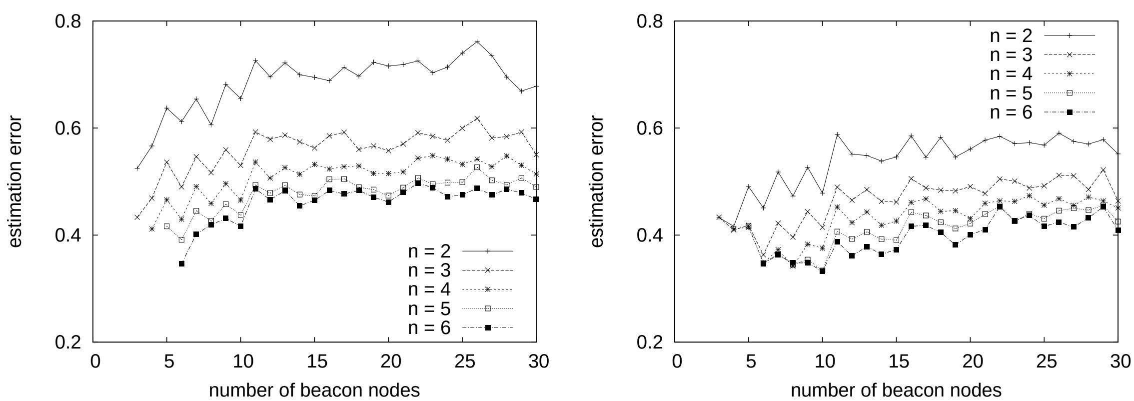
(a) IDMaps and Hotz's triangulation (b) GNP and ICS

Figure 6: Comparison of IDMaps, Hotz's triangulation, GNP, and ICS for the NLANR data set.



(a) ICS (b) GNP

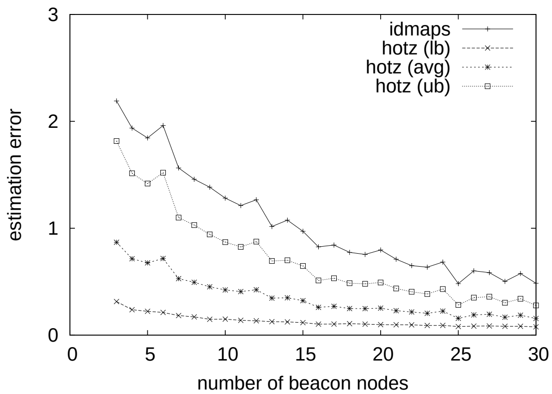
Figure 7: The effect of the dimension of the coordinate system on the performance of ICS and GNP for the NLANR data set.



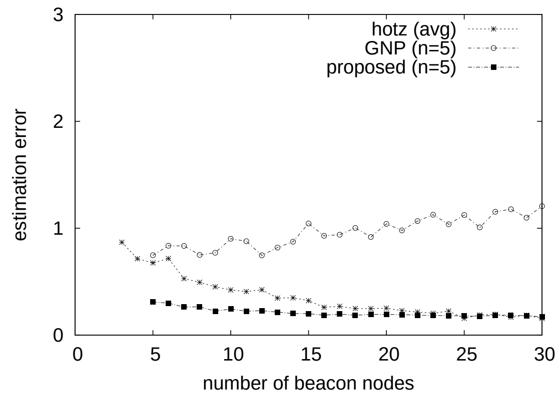
(a) $k = n$ (b) $k = 2n$

Figure 8: Performance of the partial measurement method in ICS for the NLANR data set.

Two-level hierarchical topology

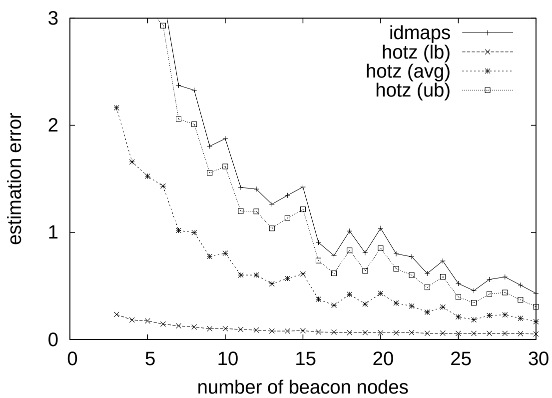


(a) IDMaps and Hotz's triangulation

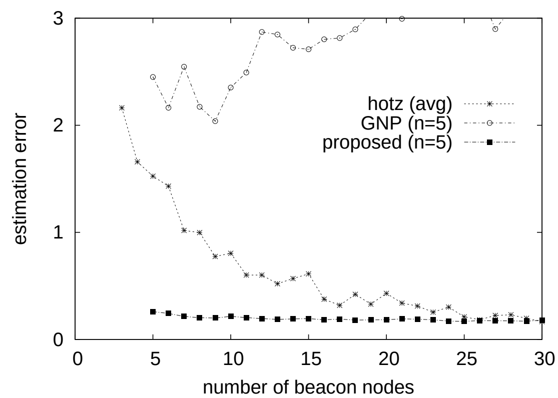


(b) GNP and ICS

Three-level hierarchical topology



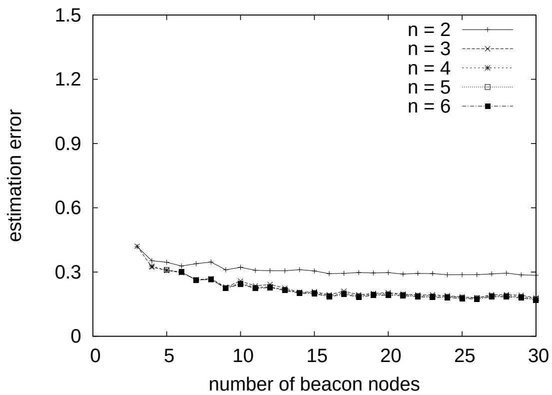
(c) IDMaps and Hotz's triangulation



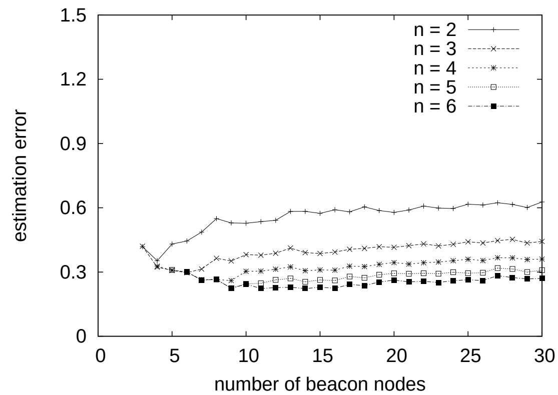
(d) GNP and ICS

Figure 9: Comparison of IDMaps, Hotz's triangulation, GNP, and ICS for the GT-ITM data set.

Two-level hierarchical topology

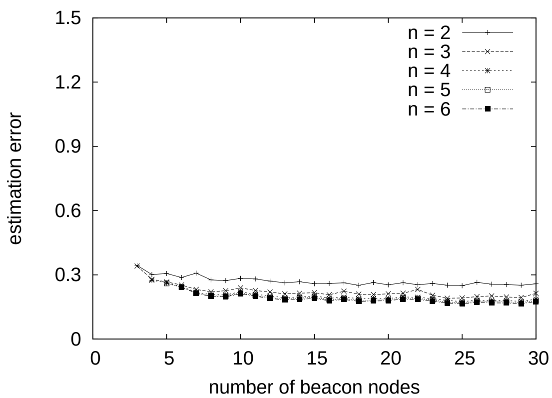


(a) Full measurement

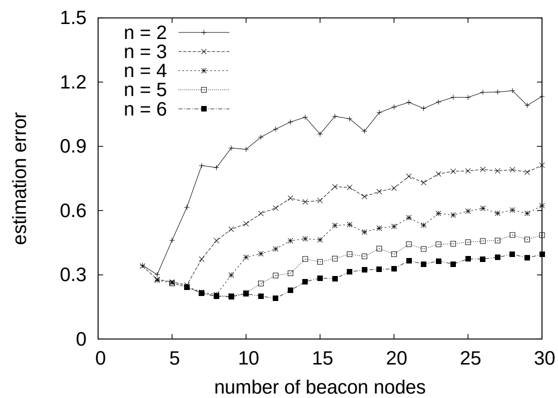


(b) Partial measurement $k = 2n$

Three-level hierarchical topology



(c) Full measurement



(d) Partial measurement $k = 2n$

Figure 10: Effect of the dimension of the coordinate system on the performance of ICS for the GT-ITM data set.

become larger for the NLANR data set as m increases. As compared with the two bounds of Hotz's triangulation, the average of the two bounds is less sensitive to the number of beacon nodes. GNP can estimate distances accurately only when the number of variables in the optimization problems is small, i.e, the numbers of the beacon nodes and the coordinate dimension are small. ICS provides accurate estimates under most cases, regardless of the number of beacon nodes, the coordinate dimension, and the level of topology complexity. ICS with the partial measurement method reduces the number of measurements, while its accuracy is not significantly degraded when the number of beacon nodes and the coordinate dimension are large.

7. CONCLUSION

In this paper, we present a new Coordinate system called the *Internet Coordinate System (ICS)*. We show that the principal component analysis (PCA) technique can effectively extract topological information from delay measurements between beacon hosts. Based on PCA, we devise a transformation method that projects the distance data space into a new coordinate system of (much) smaller dimensions. The transformation retains as much topological information as possible and yet enables end hosts to easily determine their locations in the coordinate system. We show via experiments using both real measured and synthetic data sets that ICS can make accurate and robust estimates of network distances between end hosts, regardless of the number of beacon nodes and the level of network topology complexity. Finally, we also show the number of measurements made by a host can be reduced without much loss of accuracy.

We have identified several research avenues. In particular, we will investigate several clustering algorithms to be used by the administrative node in the partial measurement method (*M2*) so as to facilitate selection of median beacon nodes. We will also study in whether or not, and to what extent, placement of beacon nodes has a significant impact on the PCA-derived coordinate system in measuring network distances.

8. REFERENCES

[1] P. Francis, S. Jamin, V. Paxson, L. Zhang, D. F. Gryniewicz, and Y. Jin, "An architecture for a global Internet host distance estimation service," in *Proceedings of IEEE Infocom*, 1999.

[2] E. Ng and H. Zhang, "Predicting Internet network distance with coordinates-based approaches," in *Proceedings of Infocom*, 2002.

[3] S. Hotz, *Routing information organization to support scalable interdomain routing with heterogeneous path requirements*, Ph.d. thesis, Univ. of Southern California, 1994.

[4] J. D. Guyton and M. F. Schwartz, "Locating nearby copies of replicated Internet servers," in *Proceedings of ACM Sigcomm*, 1995.

[5] L. Tang and M. Crovella, "Virtual landmarks for the Internet," in *Proceedings of ACM Internet Measurement Conference*, 2003.

[6] T. H. Cormen, C. E. Leiserson, and R. L. Rivest, *Introduction to Algorithms*, MIT Press, 1990.

[7] S. Ratnasamy, M. Handley, R. Karp, and S. Shenker, "Topologically-Aware overlay construction and server selection," in *Proceedings of Infocom*, 2002.

[8] J. A. Nelder and R. Mead, "A simplex method for function minimization," *Computer Journal*, vol. 7, pp. 308–313, 1965.

[9] I. T. Jolliffe, *Principal component analysis*, New York: Springer-Verlag, 1986.

[10] B. Noble and J. W. Daniel, *Applied Linear Algebra*, Prentice Hall, 1988.

[11] K. Y. Yeung and W. L. Ruzzo, "Principal component analysis for clustering gene expression data," *Bioinformatics*, vol. 17, no. 9, pp. 763–774, 2001.

[12] C. Ding, X. He, H. Zha, and H. Simon, "Adaptive dimension reduction for clustering high dimensional data," in *Proceedings of the 2nd IEEE Int'l Conf. Data Mining*, 2002, pp. 147–154.

[13] T. P. Minka, "Automatic choice of dimensionality for PCA," in *Technical report 514, MIT Media Laboratory*, 2000.

[14] V. Paxson, "End-to-end routing behavior in the Internet," in *Proceedings of SIGCOMM '96*, August 1996.

[15] National laboratory for applied network research, "Active measurement project (AMP)," http://watt.nlanr.net/.

[16] A. Jain and R. C. Dubes, *Algorithms for clustering data*, Prentice Hall, 1988.

[17] E. W. Zegura, K. Calvert, and S. Bhattacharjee, "How to model an Internetwork," in *Proceedings of IEEE Infocom*, 1996.

[18] J. Fall and K. Varadhan, "ns manual," http://www.isi.edu/nsnam/ns/ns-documentation, 2001.