Non-metric coordinates for predicting network proximity

Peter Key
Microsoft Research
Cambridge, UK
peter.key@microsoft.com

Laurent Massoulié
Thomson
Paris Research Lab, France
laurent.massoulie@thomson.net

Dan-Cristian Tomozei
Thomson
Paris Research Lab, France
dan-cristian.tomozei@m4x.org

***Abstract*—We consider the problem of determining the "closest", or best Internet host to connect to, from a list of candidate servers. Most existing approaches rely on the use of metric, or more specifically Euclidean coordinates to infer network proximity. This is problematic, given that network distances such as latency are known to violate the triangle inequality. This leads us to consider non-metric coordinate systems. We perform an empirical comparison between the "min-plus" non-metric coordinates and two metric coordinates, namely L-infinity and Euclidean. We observe that, when sufficiently many dimensions are used, min-plus outperforms metric coordinates for predicting Internet latencies.**

We also consider the prediction of "widest path capacity" between nodes. In this framework, we propose a generalization of min-plus coordinates. These results apply when node coordinates consist in measured network proximity to a random subset of landmark nodes. We perform empirical validation of these results on widest path bandwidth between PlanetLab nodes.

We conclude that appropriate non-metric coordinates such as generalized min-plus systems are better suited than metric systems for representing the underlying structure of Internet distances, measured either via latencies or bandwidth.

I. Introduction

We consider the problem of determining the *closest* or *best* Internet host to connect to, from a list of candidate servers, by using *network coordinates* instead of direct measurements.

In one application of interest, closeness is measured by IP packet round-trip time latencies, and several candidate DNS servers are proposed to a host. The latter then approximates its latency to a particular server by computing a pseudo-distance between its and the server's coordinates. Thus the decision can be made solely from the coordinates of hosts and servers, without resorting to direct measurements.

In another application, a node participating in a peer-to-peer network aims to enter the overlay by connecting to the nodes with the largest available uplink bandwidth. Similarly, it can make its decision based on coordinates rather than direct measurements.

The growth of peer-to-peer applications and location-aware services motivates the search for efficient solutions in both contexts.

Many approaches (see e.g. [8]–[10]) have been proposed for predicting Internet latencies. These typically use metric, or more specifically Euclidean coordinates. On the other hand, recent work by Kleinberg et al. [5] established theoretical guarantees on the quality of prediction by non-Euclidean metric coordinates, and by particular non-metric coordinates.

To be more specific, we introduce the following notation. We consider a finite set of N nodes, indexed by $i = 1, \dots, N$, and the problem of predicting some function d_{ij} of two nodes i, j, as a function $f(x_i, x_j)$. Function d_{ij} is typically symmetric, but not necessarily a distance. The coordinate vectors x_i of node i are to be inferred; ideally few coordinates are used.

In this setup, Kleinberg et al. considered both the L^∞ distance function $f_{L^\infty}(x_i, x_j) := \sup_{\ell=1}^{d} |x_i(\ell) - x_j(\ell)|$, and the so-called *min-plus* function, $f_{min-plus}(x_i, x_j) := \min_{\ell=1}^{d}(x_i(\ell)+x_j(\ell))$. For both functions, they showed that the corresponding prediction is close to the original distance d_{ij}, for most pairs (i, j), provided the original function d_{ij} is a well-behaved metric (more specifically, it has small *doubling dimension*).

The coordinates $x_i(\ell)$ used in [5] are in fact measurements between node i and some *landmark node* z_ℓ, which is selected at random from the whole collection of nodes. Thus these coordinates are obtained in one shot, from d measurements per node. In contrast the metric coordinates used in [8]–[10] are continuously updated on the basis of new measurements, and thus potentially more finely tuned than in a one-shot approach.

The applicability of these results to Internet latencies is however not straightforward: it is well known that Internet latencies violate the triangle inequality, e.g. [12], [20]. Hence they do not constitute a metric, and a fortiori not a metric with bounded doubling dimension.

The problem of bandwidth prediction based on network coordinates has received far less attention; in this context there is no particular reason to think that metric coordinates would be appropriate.

Our aim in this paper is to identify non-metric coordinate systems for accurate prediction of either Internet latencies, or bandwidth.

Our contributions are as follows:

- We establish that min-plus coordinates can represent exactly any symmetric, not necessarily metric function d_{ij}. This motivates the use of min-plus coordinates as a "universal" system, even in non-metric environments.
- We compare the performance of Vivaldi-like "Euclidean + height" coordinates to L^∞ and min-plus

coordinates, using continuous refinements for all three. The comparison is performed on three data sets, namely S3, PlanetLab and King. We observe good performance for all three, but min-plus coordinates benefit more from additional coordinates.

- We then consider predicting widest-path bandwidth measures. We validate these results empirically on the prediction of widest path bandwidth between nodes in the S3 data set. In an extended version of the paper, we introduce a generalization of min-plus coordinates to this context and we prove theoretical bounds on the corresponding performance for "one-shot" coordinates that are measurements to randomly selected landmarks.

II. Min-plus vs metric coordinates for latency prediction

A. The case for min-plus coordinates

Consider a set of N nodes, indexed by $i = 1, \dots, N$, and a function $d_{ij} : N \times N \to \mathbb{R}$ of pairs of nodes. It has been known since Fréchet [3] that, when d_{ij} is a metric, one can assign N coordinates $x_i(1), \dots, x_i(N)$ to each node i, in such a way that the L^∞ distance between the coordinate vectors x_i and x_j coincides with d_{ij}. Indeed, this holds by setting $x_i(k) = d_{ik}$.

In practice we want to use far fewer than N coordinates. Nevertheless, the fact that an exact reconstruction of metric d_{ij} can be done with many coordinates in the L^∞ metric makes it a good candidate for predicting such d_{ij}, even with few coordinates. Indeed, one might hope that reconstruction accuracy degrades gracefully as the number of dimensions is reduced.

We now establish that a similar result holds for min-plus coordinates, and for arbitrary symmetric, not necessarily metric d_{ij}:

Lemma 1: Any symmetric function d_{ij} on a set of N points can be represented as a min-plus distance, i.e.

$$d_{k\ell} = f_{min-plus}(x_k, x_\ell) = \min_i (x_k(i) + x_\ell(i)), \ k \neq \ell \quad (1)$$

irrespective of whether it satisfies the triangle inequality or not, for suitable N-dimensional coordinate vectors x_i, $i = 1, \dots, N$.

Proof: We use the following explicit construction. For $i \in \{1, \dots, N\}$, and each $k \in \{1, \dots, N\}$, $k \neq i$, we set the i-th coordinate $x_k(i)$ of point k to

$$x_k(i) = d_{ik} - x_i(i). \quad (2)$$

The i-th coordinate of point i, that is $x_i(i)$ is then set to

$$x_i(i) = \frac{1}{2} \min_{k,\ell \in \{1,\dots,N\}} (d_{ik} + d_{i\ell} - d_{k\ell}). \quad (3)$$

We now check that (1) holds. For a pair of points k, ℓ, by (2) we have that $x_k(\ell) + x_\ell(\ell) = d_{k\ell}$, so that

$$\min_{i=1}^{N} (x_k(i) + x_\ell(i)) \leq d_{k\ell}.$$

On the other hand, for $i \neq k$, $i \neq \ell$, by (2) and (3) we have that

$$\begin{aligned} x_k(i) + x_\ell(i) &= d_{ik} + d_{i\ell} - 2x_i(i) \\ &\geq d_{ik} + d_{i\ell} - (d_{ik} + d_{i\ell} - d_{k\ell}) \\ &= d_{k\ell}. \end{aligned}$$

This concludes the proof. ■

Note that no claim is made about prediction of the quantities d_{ii}: the construction does not necessarily satisfy $d_{ii} \geq x_i(i) + x_i(i)$.

This result suggests that min-plus coordinates can be useful for predicting non-metric quantities d_{ij}, such as Internet Round Trip Times, as documented in [6].

B. Algorithms for setting network coordinates

We now describe the scheme we use for setting coordinates in our empirical comparison. It is inspired by the Vivaldi system design [8]. The basic algorithm in Vivaldi is a distributed optimisation, based on modelling the network as a spring-connected system, and letting the system evolve to a minimum energy state. More precisely, nodes continuously contact one another; when node i contacts node j, it performs a small adaptation of its coordinate vector x_i so as to reduce the absolute difference between the actual quantity d_{ij} to be predicted, and the prediction $f(x_i, x_j)$. This can be seen as an approximation of a gradient descent algorithm for minimizing the so-called stress metric, defined as

$$\sigma_r = \sum_{i<j} (d_{ij} - f(x_i, x_j))^2. \quad (4)$$

Stress (and variants with distinct normalizations) is a common measure of prediction accuracy in the applied statistics literature (e.g. [2])

Vivaldi [8] proposed a variant of Euclidean coordinates to better model Internet latencies, and introduced the notion of "height" parameters, namely the coordinate x_i decomposes into y_i, h_i where y_i is $d-1$-dimensional, and h_i is a nonnegative height. The scheme then uses the prediction

$$f_{Vivaldi}(x_i, x_j) = ||y_i - y_j||_2 + h_i + h_j,$$

where $||y_i - y_j||_2 = \sqrt{\sum_\ell (y_i(\ell) - y_j(\ell))^2}$. In what follows we relax the non-negativity constraint on heights, as we observed that this improves performance.

For our experiments, we use a distributed algorithm inspired by Vivaldi, adapted to each specific coordinate system.

The Algorithm for min-plus coordinates minimizes the same stress cost function in a distributed fashion. Each node i updates its coordinates and error estimates when it obtains a real RTT measurement for node j. The updates are based on both the measured RTT, the coordinates of node j and an *error estimate* communicated by node j. The details for the case of min-plus coordinates are described in Algorithm 1 below.

Updates for coordinates and errors use simple weighted moving averages. The gain for coordinate updates is wc_c, where w is defined in the Algorithm, and the factor c_c has been selected experimentally.

The algorithms used for Vivaldi and L^∞ coordinates were similar, using a suitably modified definition for f_{ij}.

C. Measures of Accuracy

Several performance indices could be used to measure the accuracy of particular prediction schemes, the "stress"

Algorithm 1 Algorithm for node i, using min-plus coordinates

```
RPC-MIN-PLUS-UPDATE($rtt, x_j, e_j$)
// Node $j$ is $rtt$ ms away, has coordinates $\mathbf{x_j}$
// and error estimate $e_j$
$f_{ij} \leftarrow \min_{l=1}^{d}(x_i(l) + x_j(l))$
$w \leftarrow \frac{e_i}{e_i+e_j}$
$e_s \leftarrow \frac{|f_{ij}-rtt|}{rtt}$ {Relative error}
$e_i \leftarrow e_s \cdot c_e \cdot w + e_i \cdot (1 - c_e \cdot w)$
if $f_{ij} < rtt$ then
    for all $l$ such that $x_i(l) + x_j(l) = f_{ij}$ do
        $x_i(l) \leftarrow x_i(l) + c_c \cdot w \cdot (rtt - f_{ij})$
    end for
else
    Choose at random $l$ such that $x_i(l) + x_j(l) = f_{ij}$
    $x_i(l) \leftarrow x_i(l) - c_c \cdot w \cdot (f_{ij} - rtt)$
end if
```

function being one possible objective function. An alternative objective function, namely the *relative rank loss* (RRL), is introduced in [6] as a better index for the present application context of "best node selection". RRL is defined as a function of the actual quantities d_{ij} and their estimates $\widehat{d_{ij}}$ as follows. It counts the fraction of ordered triplets (i, j, k) for which the distances d_{ij} and d_{ik} are ordered in the opposite way as $\widehat{d_{ij}}$ and $\widehat{d_{ik}}$. This cost function can be interpreted as the probability that some node i fails to select the closest neighbor among two candidates j, k when relying on comparison of reconstructed rather than actual distances.

We propose a variant of this measure of accuracy, a *Smooth Relative Rank Loss*, SRRL, which has two additional features: (a) it puts no penalty when the mis-ordered nodes j, k have distances d_{ij}, d_{ik} to the querying node i that differ by no more than some absolute tolerance d_{min} (typically 10ms); (b) it puts a penalty of no more than $\min(1, (d_2 - d_1)/d_1)$ when the swapped distances are d_1 and d_2, $d_1 < d_2$. The meaning is that the penalty is bounded by the relative error $(d_2 - d_1)/d_1$ made in choosing the node at distance d_2.

The explicit of the SRRL is given by

$$\sigma_{srrl} = \frac{1}{N(N-1)(N-2)} \sum_{i,j,k: i \neq j \neq k} \varphi(d_{ij}, d_{ik}) \cdot \mathbf{1}_{\widehat{d}_{ij} < \widehat{d}_{ik}} \tag{5}$$

where φ is the cost function associated with a misordering,

$$\varphi(d_2, d_1) = 1 - \exp\left[-\max\left(0, \frac{d_2 - d_1 - d_{\min}}{d_1 + d_{\min}}\right)\right]. \tag{6}$$

These minor modifications we brought to the RRL cost function are practically motivated: SRRL is tolerant to small relative as well as absolute errors. It is "smoother" than RRL and therefore easier to manipulate.

D. Experimental Framework

We compared the results obtained for three data sets, PlanetLab [17], HP S3 [16], King [18]. Each data set comprises latency or bandwidth measurements between all node pairs, expressed as matrices. These matrices were input into a network packet simulator and used for testing the various algorithms. Although the algorithms themselves are distributed, and only require a small subset of all the measurements per iteration, the full matrix is used to compute the resulting SRRL.

The *PlanetLab* data set was obtained by parsing the All Pair Ping tool logs. The logs were generated during three months – February, March and April 2006 – with measurements performed every 20 minutes. For each pair of nodes, the smallest reported latency was taken as the input to the final distance matrix, as it best reflects the actual physical properties of the IP path between the nodes. Missing values in the resulting matrix were filled by computing a shortest path distance over the graph with edge lengths set to the smallest reported latency, and to infinity in the absence of observations. Note that for node pairs with observed latencies, the smallest observed latency was not replaced by a shortest path distance.

The largest connected component of the resulting graph has 217 nodes. The triangle inequality is violated for 21.18% of all the triplets (i, j, k) from this largest component.

The *S3* contains latency and bandwidth information. The latency is estimated via the HP tool "NetVigator". For latency measurements, we consider the largest connected component of the corresponding graph. This gives a 419-node graph with almost no missing edges. The triangle inequality violation ratio equals 9.09%.

For bandwidth measurements (used in Section IV), we used a "widest-path" algorithm to fill in the high percentage of missing edges (again, by leaving the existing values unmodified). The analogue of the triangle inequality is violated for 78.64% of the triangles. This was expected, as Internet routing is not designed for finding widest paths.

The *King* data set involves 1740 Internet DNS servers. The technique used to obtain this data is described in [11]. The RTT between servers A and B is approximated by the difference between the RTT measured to A and the RTT measured to B, going through A[1]. The resulting matrix of RTT's is available online at [18]. For this data set, the triangle inequality violation ratio equals 12.68%.

E. Results

We compare the performance of schemes according to the Smooth Relative Rank Loss (SRRL) criterion. Figures 1, 3 and 2 show the SRRL for varying dimensions and for the data sets S3, King and Planet Lab respectively. In all three cases, we compare the performance of "Vivaldi" (Euclidean + height), L^{∞} and min-plus coordinates. In all cases, the coordinates are optimized to minimize stress (4) using the corresponding modified Vivaldi algorithm[2]. The

[1] a DNS query is sent to A for a domain served by B, to ensure that the data packet will follow the path A – B – A.

[2] It may seem more consistent to perform continuous optimization of the actual criterion of interest, namely the SRRL. We have done such experiments, using a regularized version of SRRL, the gradient of which is well defined. The corresponding simulation results are very similar, with achieved SRRL's only marginally smaller.

algorithm runs until it reaches a stable state. We observed empirically that less than 15000 exchanges suffice. The result depends marginally on the actual random nodes that each node contacts to update its own coordinates. This is reflected by the empirical standard deviations for the resulting average SRRL, represented on the plots as error bars (± standard deviation). For the King data set, we only evaluated "Vivaldi" and min-plus coordinates, and for only up to 20 dimensions, since simulations are particularly long for this data set.

We stress that it is reasonably cheap in practice to use many coordinates: the impact on local storage at nodes is manageable; the number of communications between nodes is unaffected; for each communication, more coordinates need to be exchanged, but the amount of coordinates that can be communicated in a single IP packet is well above 100.

For both S3 and PlanetLab, we observe that min-plus coordinates outperform Vivaldi coordinates for dimensions above 18 and 12 respectively, in terms of SRRL (Figures 1 and 2). We also noticed that, in terms of RRL (not reported here due to lack of space), 8 and 4 dimensions respectively suffice for min-plus coordinates to outperform Vivaldi coordinates. L^{∞} coordinates achieve an intermediate performance. For lower dimensions, Vivaldi coordinates typically outperform the other two. More importantly, we observe that the reported SRRL are small even for as few as three dimensions. Min-plus appears better at exploiting additional dimensions.

The King data set yields a different picture, with Vivaldi coordinates outperforming min-plus coordinates (Figure 3). But again, the most striking fact is perhaps that small SRRL can be achieved with few dimensions, and this for all proposed coordinate systems.

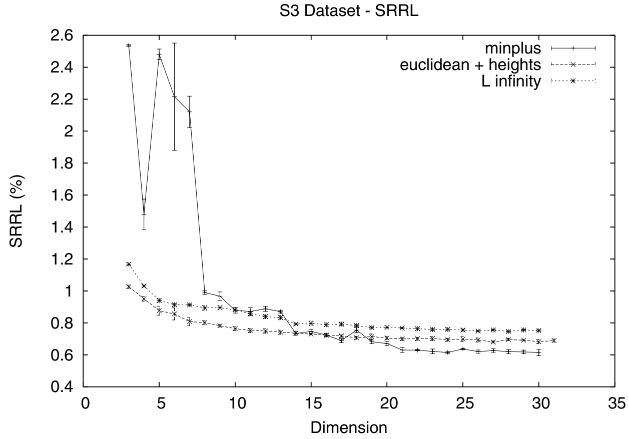

Fig. 1. Smooth Relative Rank Loss - S3 data set (6 runs per point)

III. Generalisation of min-plus coordinates

In this section, we give an intuitive generalisation of the min-plus coordinates applied to bottleneck bandwidth prediction in a widest path routing context.

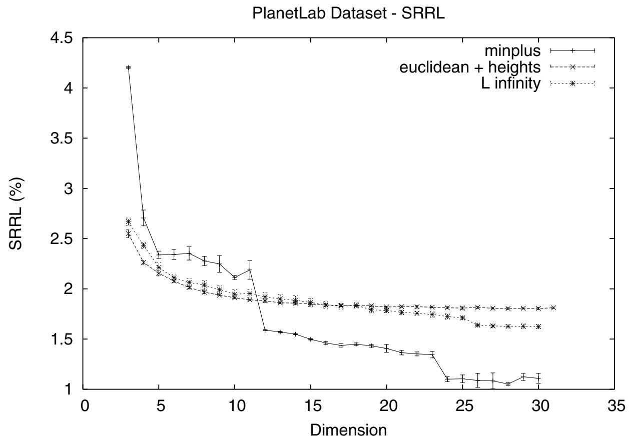

Fig. 2. Smooth Relative Rank Loss - PlanetLab data set (10 runs per point)

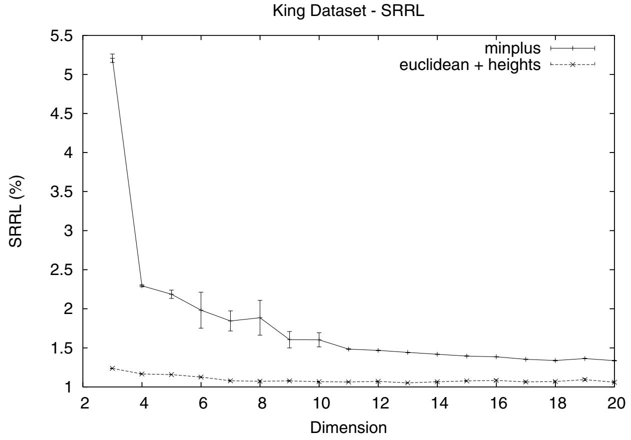

Fig. 3. Smooth Relative Rank Loss - King data set (4 runs per point)

Given a set of landmark nodes $\ell(1), \ldots, \ell(K)$ and $w_{i\ell(1)}, \ldots, w_{i\ell(K)}$ for all nodes i, we estimate the weight w_{ij} as the max-min distance:

$$\hat{w}_{ij} = \max_{k=1}^{K} \left(\min(w_{i\ell(k)}, w_{j\ell(k)}) \right). \qquad (7)$$

In this case, we can show that by choosing sufficiently many landmarks we can ensure that with high probability $\hat{w}_{ij} = w_{ij}$, so there is *no* distortion. Hence perfect reconstruction is possible for almost all bottleneck capacities. Further details are available in an extended version of the paper.

Figure 4 shows the RRL (in percentage) based on the extended min-plus scheme (or equivalently, max-min scheme) described in Section III, based on a random selection of landmarks. The RRL, defined by analogy with that of Section II-C, counts the fraction of ordered triples (i, j, k) for which bottleneck capacities w_{ij} and w_{ik} are ordered in the opposite way to max-min esimates $\hat{w}_{ij}$ and $\hat{w}_{ik}$. The curve labeled "Original S3 data" is obtained using the original bandwidths reported in S3; we see that RRL remains at 30% irrespective of the number of dimensions. The curve labeled "Widest path S3 data" is obtained from the modified S3 data, where original bottleneck bandwidth measures have been replaced by widest path bandwidth measures. We see that for 2 dimensions, the achieved RRL is already below 0.1%, and is further decreased by additional dimensions. This confirms the theoretical result of section III, stating that the proposed scheme allows *exact* reconstruction of bottleneck bandwidth for most node pairs, under the assumption of widest-path routing.

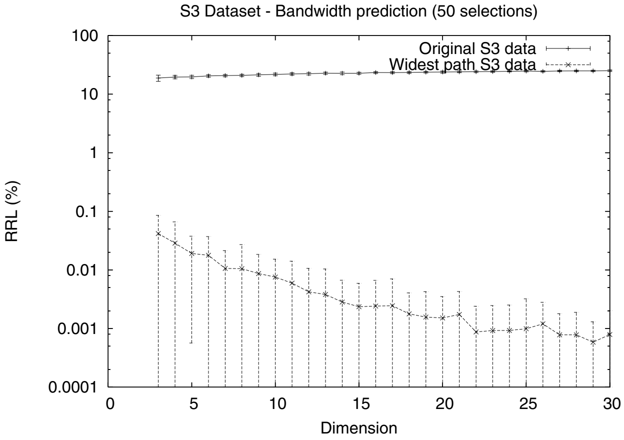

Fig. 4. Bandwidth - S3 data set

IV. Conclusions

We have considered the problem of selecting the best host to connect to, using network coordinates, and more specifically non-metric coordinates. We first showed that min-plus coordinates can represent perfectly any symmetric function d_{ij} (such as Internet latencies), if we use a large number of dimensions.

We compared the performance of min-plus, against two metric coordinate systems, namely L^∞ and Vivaldi for predicting Internet latencies on three data sets. The three approaches led to comparable – and good– performance, when individual node coordinates were finely optimized. Min-plus slightly outperformed the other two approaches when using large number of dimensions.

It is interesting to notice that coordinate systems such as min-plus and even L^∞, proposed initially purely for theoretical reasons, manage performance comparable to or better than Vivaldi. In contrast, Vivaldi uses Euclidean+heights coordinates in an attempt to capture real-world features.

Bandwidth prediction using coordinates has received little attention. We proposed an extension of min-plus coordinates to a general algebraic framework for routing, which encompasses bottleneck bandwidth prediction. In this setup, we proved (refer to extended version) performance guarantees of schemes based on random landmark selections. In particular, we proved that exact reconstruction of bottleneck bandwidth can be done with few dimensions and for most node pairs, assuming widest-path routing is used. This assumption is realistic in a peer-to-peer system which implements overlay routing.

An extended version of this paper gives further theoretical bounds in prediction quality and presents a generalised min-plus routing algebra.

References

[1] I. Abraham, Y. Bartal, T-H. Chan, K. Dhamdhere, A. Gupta, J. Kleinberg, O. Neiman and A. Slivkins. Metric embeddings with relaxed guarantees. Proc. 46th IEEE Symposium on Foundations of Computer Science, 2005.

[2] I. Borg and P.J.F. Groenen. *Modern Multidimensional Scaling*. 2nd Editions. Springer, 2005

[3] M. Fréchet. Les dimensions d'un ensemble abstrait. Math. Ann. 68 (1909-1910), 145-168.

[4] R. E. Gomory and T.C. Hu. Multi-terminal network flows. Journal of the SIAM, 9:551-570, 1961.

[5] J. Kleinberg, A. Slivkins and T. Wexler. Triangulation and Embedding using Small Sets of Beacons. Proc. 45th IEEE Symposium on Foundations of Computer Science, 2004.

[6] E.K. Lua, T. Griffin, M. Pias, H. Zheng and J. Crowcroft. On the accuracy of embeddings for Internet coordinate systems. Internet Measurement Conference (IMC), 2005.

[7] J. L. Sobrinho. Algebra and Algorithms for QoS Path Computation and Hop-by-Hop Routing in the Internet. IEEE/ACM Transactions on Networking, Vol. 10, No 4, 541-550, 2002.

[8] F. Dabek, R. Cox, F. Kaashoek, R . Morris, *Vivaldi: A Decentralized Network Coordinate System*, SIGCOMM'04, MIT CSAIL, Cambridge, MA

[9] M. Costa, M. Castro, A. Rowstron, P. Key, PIC: Practical Internet Coordinates for Distance Estimation. In *ICDCS*, Tokyo, Japan, 2004.

[10] L. Tang, M. Crovella, *Virtual Landmarks fot the Internet*, Department of Computer Science, Boston University

[11] K. P. Gummadi, S. Saroiu, S. D. Gribble, *King: Estimating Latency Between Arbitrary Internet End Hosts*, SIGCOMM IMW 2002

[12] S.Lee, Zhi-Li Zhang, S. Saha, On the suitability of embedding of Internet Hosts, in *Proc. ACM Sigmetrics*, 2006.

[13] Y. Shavitt and T. Tankel, *Big-Bang simulation for embedding network distances in Euclidean space*, E.E.-Systems Department, Tel-Aviv University, Tech. Rep., July 2002. http://citeseer.ist.psu.edu/shavitt02bigbang.html

[14] W. H. Press, S. A. Teukolsky, W. T. Vetterling, B. P. Flannery, *Numerical Recipes in C – The Art of Scientific Computing – Second Edition*, Cambridge University Press

[15] N. Hu, P. Steenkiste, *Estimating Available Bandwidth Using Packet Pair Probing*, Carnegie Mellon, CMU-CS-02-166

[16] HP's Scalable Sensing Service (S3),
http://networking.hpl.hp.com/s-cube/index.html

[17] PlanetLab All Pair Pings,
http://pdos.csail.mit.edu/~strib/pl_app/,
http://ping.ececs.uc.edu/ping/

[18] The King Data Set,
http://pdos.csail.mit.edu/p2psim/kingdata/

[19] B. Wong, A. Slivkins, E.G. Sirer. *Meridian: A Lightweight Network Location Service without Virtual Coordinates.* In *SIGCOMM*, August 2005.

[20] H. Zheng, E. K. Lua, M. Pias, T. G. Griffin, *Internet Routing Policies and Round-Trip-Times*, in *Passive and Active Measurement Workship (PAM)*, 2005.

[21] T. S. Eugene Ng, Hui Zhang, *Predicting Internet Network Distance with Coordinates-Based Approaches*, IEEE INFOCOM'02, New York, NY, June 2002