Geometric Exploration of the Landmark Selection Problem

Liying Tang and Mark Crovella

Department of Computer Science, Boston University, Boston, MA 02215
{litang,crovella}@cs.bu.edu

Abstract. Internet coordinate systems appear promising as a method for estimating network distance without direct measurement, allowing scalable configuration of emerging applications such as content delivery networks, peer to peer systems, and overlay networks. However all such systems rely on landmarks, and the choice of landmarks has a dramatic impact on their accuracy. Landmark selection is challenging because of the size of the Internet (leading to an immense space of candidate sets) and because insight into the properties of good landmark sets is lacking. In this paper we explore fast algorithms for landmark selection. Whereas the traditional approach to analyzing similar network-based configurkation problems employs the graph structure of the Internet, we leverage work in coordinate systems to treat the problem geometrically, providing an opening for applying new algorithms. Our results suggest that when employing small numbers of landmarks (5-10), such geometric algorithms provide good landmarks, while for larger numbers of landmarks (20-30) even faster methods based on random selection are equally effective.

1 Introduction

Internet coordinate schemes assign coordinate vectors to hosts in the Internet, and attempt to do so in a manner such that the Euclidean distance between hosts approximates their network distance, usually taken to be minimum RTT [7]. The power of such coordinate systems is that they allow network distance to be estimated in the absence of measurement. As a result, applications for which network distance matters (such as content delivery networks, peer to peer applications, end-system multicast, and others) can be rapidly configured in a network-sensitive manner based on static input data (*i.e.,* a database of nodes and their coordinates).

A number of proposals have been put forward for such systems, ranging from initial proposals based on expensive nonlinear optimization [7, 14] to more recent fast methods based on Lipschitz embedding and dimensionality reduction [6, 13]. All such proposals start from a set of inter-node measurements. For scalability, it is assumed that all nodes desiring coordinates cannot measure all distances of interest (*i.e.,* distances to all other nodes), and so must choose a set of *landmarks* by which to establish their coordinates. Thus, the usual arrangement is that all nodes measure their distances to a fixed set (or at least to a fixed number) of landmarks, and use those measurements to determine their coordinates, so that measurement load in the network scales linearly with the number of nodes. Early work assumed a fixed set of landmarks, which recent proposals have suggested that measurement load may be distributed by using multiple landmark sets [13, 10].

Thus, in order to deploy coordinate systems, specific landmarks must be chosen. There is reason to believe that the accuracy of Internet coordinate systems is sensitive to the particular choice of landmarks. First, larger landmark sets are generally more accurate, because they bring more information to bear on the coordinate construction step. Second, two landmarks that are "close" to each other in the Internet may provide little discriminatory power – meaurements to both landmarks will be nearly the same and will therefore add little information in the coordinate construction step.

Because of the large number of possible landmarks (hundreds of millions in the Internet) and the complex relationship between landmarks used and the accuracy of the resulting coordinate scheme, optimal landmark selection is a very challenging problem. A natural approach in seeking heuristics to guide the search is to look for structure in the distance measurements found in the Internet, and to try to place landmarks in ways that exploit that structure.

In this paper we examine the structure of Internet distance measurements, and ask whether that structure can help us select good landmarks in an efficient way. While network structure is usually throught of as a property of topology, we take a novel approach that leverages the notion of internet coordinates: we examine the *geometric* structure of network distances. We show that this structure is characterized by a high degree of clustering.

We then proceed to ask what sorts of geometric algorithms, including those explicitly making use of clusters, can be used to efficient landmark selections. We study this question using the Virtual Landmarks embedding method and find that the answer depends on the size of the landmark set. Our initial results show that when using a small number of landmarks (5-10), the proper choice of landmarks is quite important; but when a larger number of landmarks is used, even random selection is as effective as geometrically-informed approaches

Our results point the way to the ability to perform landmark selection in very large datasets (since the geometric methods are relatively scalable) and suggest that if 20-30 landmarks can be used, then landmark selection can be done quite efficiently.

2 Related Work

Internet coordinate schemes were first proposed in [7]. In this work, we use the Virtual Landmarks method for forming internet coordinates as described in [13]. The Virtual Landmarks method is based on two ideas: First, it uses a Lipschitz embedding of nodes into a high dimensionsal space. In a Lipschitz embedding, the distances to a set of landmarks are taken as the coordinates of the given node. Second, it uses dimensionality reduction via Principal Component Analysis to reduce the higher-dimensional space of Lipshitz embedding to a lower-dimensional space. In this paper, all coordinates are 7-dimensional, which was found in [13] to provide a reasonable tradeoff between low dimension and accuracy. Thus, the algorithms in this paper are concerned with selecting landmarks for the Lipschitz embedding, while all error metrics are measured based on the coordinate assignments after reducing to 7 dimensions.

The question of how to select landmarks has not been explicitly addressed in previous Internet coordinate work. However it has relevance not just for Internet coordinates, but for all methods that employ Lipschitz embeddings on round-trip times.

A number of papers have used Lipschitz embeddings to assign coordinates to hosts. In [16], the goal is to map an arbitrary host to a nearby landmark whose geographic position is known. A landmark whose Lipschitz coordinates are similar to that of the host is used as the estimated location of the host. A demographic placement approach was proposed to improve the representativeness of landmarks with respect to the hosts to be located. Like this work, they also explore placing probe machines in a geographically distributed manner, to avoid shared paths.

In [9], the authors used a Lipschitz embedding to find nearby nodes for the purpose of geolocation. The number of probe machines and locations was studied in order to improve accuracy. However, specific algorithms for landmark selection were not investigated.

Placement of other Internet resources, such as caches and mirrors, has been investigated (*e.g.,* in [8] and [3]) however these studies have not looked at geometrically inspired algorithms.

In computer vision, Lipschitz embeddings have been used to generate fast indexing schemes for images. The problem for the choice of landmarks in this setting has been studied in [2, 15]. The authors in [2] propose four methods, including minimizing average distortion, which is similar to our Greedy method. Maximum distance methods similar to those we employ have been studied in [15]. However, the results are mainly relevant to the kinds of similarities found among images and it is not clear that their conclusions would apply to the problem of Internet distances.

Finally, a number of papers have proposed scalability methods for Internet coordinate schemes based on construction of multiple landmark sets [13, 10]. Such methods assume the existence of algorithms for constructing landmark sets on the fly; the work in this paper can inform the choice of those algorithms.

3 Data

We use 3 datasets in this work, which are the same as used in [13].

NLANR AMP The NLANR Active Measurement Project [1] collects a variety of measurements between all pairs of participating nodes. Most participating nodes are at NSF supported HPC sites, and so have direct connections to the Abilene network; about 10% are outside the US. The dataset we use was collected on January 30, 2003 and consists of measurements of a 116×116 mesh. Each host was pinged once per minute, and network distance is taken as the minimum of the ping times over the day's worth of data.

Skitter The Skitter project [11] is a large-scale effort to continuously monitor routing and connectivity across the Internet. It consists of approximately 19 active sites that send probes to hundreds of thousands of targets. Targets are chosen so as to sample a wide range of prefixes spanning the Internet. Results reported in [5] suggest that about 50% of the targets in this dataset are outside the US. The dataset we used was collected in December 2002; each target was pinged approximately once per day. Network distances are the minimum ping time over 12 days. The set of targets varies among active sites; selecting the largest set of rows for which complete data is available yields a 12 × 12 symmetric dataset, and a 12 × 196,286 asymmetric dataset with 2,355,565 entries.

Sockeye Our last dataset was collected at Sockeye Networks [12]. It consists of measurements from 11 active sites to 156,359 locations. Targets were chosen through a scheme designed to efficiently explore as much of the routable Internet as possible. Each active site sent a ping to each target on an hourly basis. Network distance was taken as the minimum of all pings over a single day.

4 Algorithms

Each algorithm we consider selects a set of ℓ landmarks $\{L_i\}, i = 1, .., \ell$ taken from datasets containing n hosts $H = \{h_i\}, i = 1, ..., n$. The network distance (RTT) between h_i and h_j is denoted $d(h_i, h_j)$. Each algorithm takes as input an Internet coordinate scheme $\phi : H \rightarrow \mathbb{R}^d$ and a distance function $\delta : \mathbb{R}^d \rightarrow R$. In this paper ϕ represents the mapping obtained from the Virtual Landmarks embedding into Euclidean space with $d = 7$, and δ is the Euclidean norm.

Greedy The most expensive approach is the *Greedy* algorithm [2]. The Greedy algorithm chooses the set of landmarks that minimizes the average distortion of distances at each step. We define the relative error function $Q(h_i, h_j, \phi)$ for a pair of hosts h_i, $h_j \in H$ as

$$Q(h_i, h_j, \phi) = \frac{\delta(\phi(h_i), \phi(h_j)) - d(h_i, h_j)}{d(h_i, h_j)}$$

The accuracy of the embedding ϕ can be calculated by the *average absolute error:*

$$\bar{Q}(\phi) = 2/(n^2 - n) \sum_{i>j} |Q(h_i, h_j, \delta)|$$

Exact minimization of $\bar{Q}(\phi)$ is computationally infeasible for large n and ℓ. A stepwise approach that is tractable, but still quite expensive, is the Greedy algorithm, which proceeds iteratively. The first landmark L_1 is chosen at random. Subsequent landmarks are chosen so as to give the lowest average absolute error when used together with the already-chosen landmarks. That is, in step m we choose L_m so as to minimize $\bar{Q}(\phi)$ when used with landmark set $L_1, ..., L_m$.

K-means The *k-means* algorithm is a geometric algorithm that operates on distances as computed using an Internet coordinate scheme. K-means finds ℓ disjoint clusters in the data [4]. It starts by choosing ℓ hosts to act as centroids $(c_1, ..., c_\ell)$ at random. Each centroid c_j has an associated cluster G_j. The algorithm then repeatedly performs the following three steps:

1. Assign each host $h \in H$ to the cluster G_j with the nearest centroid c_j.
2. For each cluster G_j, calculate a new center $c = \sum_{h \in G_j} \frac{h}{|G_j|}$
3. Assign a new centroid c_j to G_j as the host nearest to this center c.

These steps are iterated until the centroids and the clusters become stable. Then, these ℓ centroids chosen by the above k-means algorithm are assigned as the landmarks.

Maximum Distance The *Maximum Distance* algorithm is also a geometric algorithm. It attempts to find the maximally-distributed landmarks from the set of n hosts, based on the intuition that landmarks must be spread far apart to give useful location information [15]. The ℓ landmarks are chosen in an iterative manner. The first landmark L_1 is chosen from the set H at random. In iteration m $(1 < m \leq \ell)$ the distance from a host h_i to the set of already chosen landmarks $L_1, ..., L_{m-1}$ is defined as the $\min_{L_j} \delta(\phi(h_i), \phi(L_j))$. The algorithm selects as landmark L_m the host that has the maximum distance to the set $L_1, ..., L_{m-1}$.

Random The *Random* method is the most efficient and simplest among all four algorithms. The ℓ landmarks are randomly chosen from the n hosts.

5 Clusters

One reason for considering geometric algorithms for landmark placement is that Internet hosts show a high degree of clustering. In this section we explore empirically this clustering in our datasets.

As described in Section 2 our data consists of Internet hosts whose coordinates are points in $\mathbb{R}^n$ with $n = 7$. We can start to examine the evidence for clustering in our datasets by looking at projections of the points onto various subspaces. When using the Virtual Landmarks method for coordinate assignment, coordinates are assigned in order of significance. The first coordinate captures the largest amount of variation possible in a single dimension, the next coordinate captures the largest amount of variation in the remaining dimensions, and so on [13]. Thus the most significant projections of the resulting data are onto the subspaces spanned by the initial axes of the coordinate system.

In Figure 1(a) we show the projection of the Skitter dataset onto the space spanned by the first two axes. In Figure 1(b) we show the corresponding plot for the Sockeye dataset. To reduce plot file size we plot a random sample of 30,000 hosts from each of our datasets; plots of the entire dataset show exactly the same features.

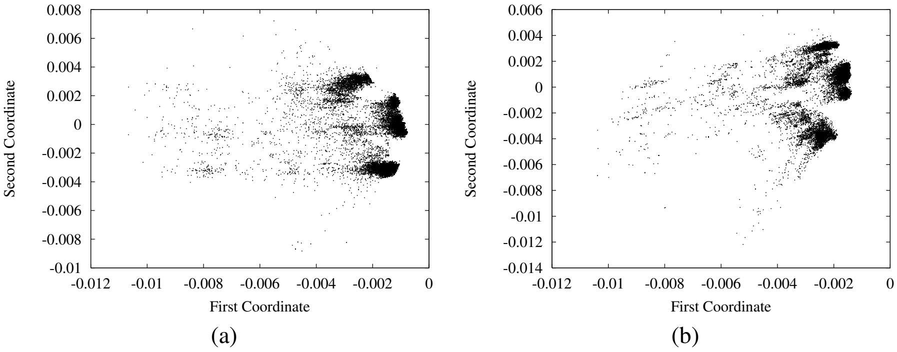


Fig. 1. Clusters in (a) Skitter Data and (b) Sockeye Data

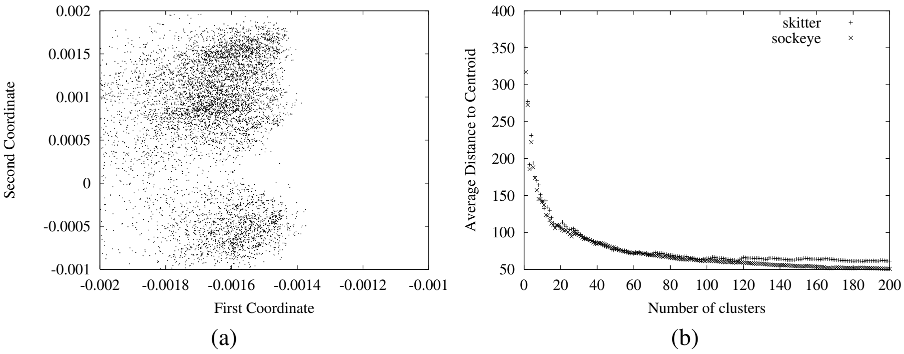


Fig. 2. (a) Zoom In on Clusters in Sockeye Data (b) Average Distance to Cluster Centroid as a Function of Number of Clusters

These plots show the remarkable amount of clustering (as well as other types of structure) that is present when Internet hosts are embedded in Euclidean space. Although the two plots represent data that was collected at different times from completely different measurements infrastructures (having different probes and target locations), they show very similar structure. The number, sizes, and relative positions of the largest clusters in each dataset are very similar.

Although these two datasets we collected in different ways, the underlying data collection strategy in each case was to try to sample all portions of the space of live Internet addresses. Thus we interpret the similarity between Figures 1(a) and 1(b) as evidence that the structure exhibited reflects real properties of Internet host connectivity.

We can also observe that clustering is present on smaller scales as well. Figure 2(a) zooms in on a part of Figure 1(b) corresponding to the largest visible cluster. This figure shows that even within a cluster, host density varies across different regions, suggesting the presence of clustering at multiple scales.

To assess the number of clusters present in our data, we adopted the following approach. Using the k-means clustering algorithm as described in Section 4, we constructed clusters for varying values of k and measured the mean cluster diameter. The resulting curves for the skitters and sockeye datasets are shown in Figure 2(b). This figure shows an inflection point around 20 clusters, indicating the presence of at least 20 clusters in both datasets.

It is very likely that these clusters are influenced by the geographical location of hosts. This can be seen in Figure 3 where we have identified the clusters formed with $k = 6$ for the AMP dataset. Although the clustering algorithm uses no direct information about geographical location, it produces clusters that are in fact geographically distinct.

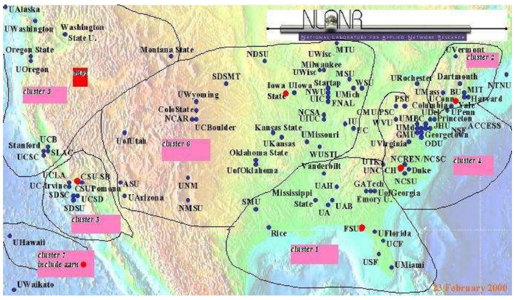


Fig. 3. Clusters in AMP data (based on graphic from [1]).

6 Algorithms for Landmark Selection

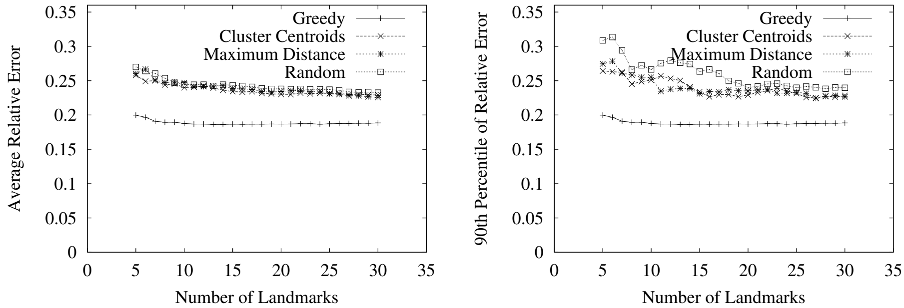
Motivated by these geometric considerations, in this section we evaluate algorithms for landmark selection.

We average absolute relative error as our principal metric. That is, for each algorithm for landmark selection, leading to an embedding ϕ, we report $\bar{Q}(\phi)$.

In addition, the entire distribution of relative error is of interest. In particular it is important to know the variability of relative error for various landmark selection methods. For these reason we also report the 90th percentile of relative error.

In applying the algorithms for landmark selection we proceed as follows. In the case of the AMP dataset, we select ℓ landmarks from the 116 hosts using each of the algorithms. We then evaluate the relative error of the Virtual Landmarks embedding of the remaining $116 - \ell$ hosts. In the case of the Skitter and Sockeye datasets, we subsample 2000 hosts at random from the set of measurement targets. We then select ℓ landmarks from this set, and evaluate the relative error of the Virtual Landmarks embedding on the active sites. For Skitter, this means we are evaluating the relative error over 12 hosts, and for Sockeye, 11 hosts. Although these sample sizes are small, the consistency of our results suggests that they are reasonably representative.

In Figures 4, 5, and 6 we present our main results. On the left in each figure is the average absolute relative error, and on the right is the 90th percentile of absolute relative error.



(a) (b)

Fig. 4. (a) Average Absolute Relative Error and (b) 90th Percentile of Relative Error of Landmark Selection Algorithms for AMP Data

In the case of the AMP hosts (Figure 4), the Greedy algorithm performs distinctly better than the others, regardless of the number of landmarks. The Random algorithm, and the geometric algorithms (K-means and Maximum Distance) have very similar performance, although the worst-case performance of Random is slightly poorer than the other two.

These results may be understood in light of the nature of the AMP hosts, which are generally sites in North America, with good connections to the high-speed Abilene network. For these hosts, the particular choice of landmark set is not too critical, although an expensive (Greedy) approach yields some benefits.

The situation is rather different when looking at lardmark sets that span the Internet as a whole, with more heterogeneous connectivity (Skitter and Sockeye datasets in Figures 5 and 6). Here the Greedy algorithm is still best, as expected, but there is a distinct difference between the others. Maximum Distance performs rather poorly, possibly because the landmarks it finds, while far from most hosts, have relatively little path diversity to the set of hosts, and therefore provide poor location information. K-means performs well even when using only a small set of landmarks. The performance of Random is poor when using a small set of landmarks, but surprisingly, it is comparable to k-means when using a large set of landmarks. This effect is especially strong when the metric of interest is the 90th percentile of absolute relative error.

These results suggest the following conclusions. First, the best-performing approach is always the expensive Greedy algorithm; if resources permit, this is the best algorithm. However, if a more efficient algorithm is needed, then for a small number of landmarks (5-10) the geometrically-based k-means algorithm is best; however, an even simpler and faster option is to expand the larkdmark set (to approximately 20-30 landmarks) and simply use Random selection.

7 Conclusions

In summary, we have advocated a geometric approach to Internet location analysis, and based on that we have described and evaluated useful algorithms for landmark

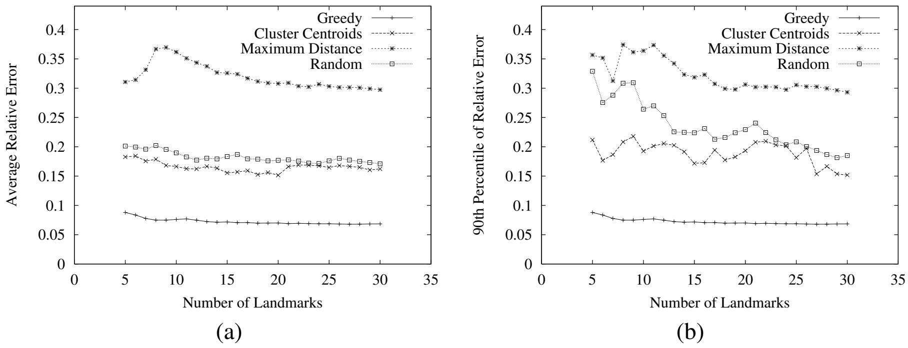


Fig. 5. (a) Average Absolute Relative Error and (b) 90th Percentile of Relative Error of Landmark Selection Algorithms for Skitter Data

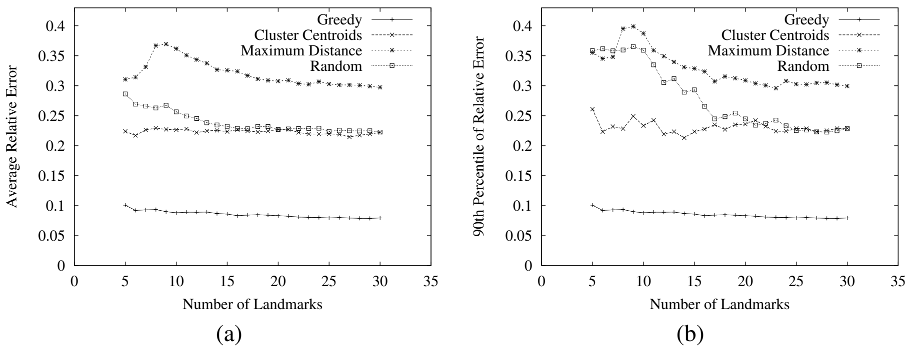


Fig. 6. (a) Average Absolute Relative Error and (b) 90th Percentile of Relative Error of Landmark Selection Algorithms for Sockeye Data

selection. Motivated by the evidence of significant clustering among Internet hosts, we explored algorithms based on k-means and maximum distance, and compared them to greedy and random alternatives. We find that the greedy algorithm is best in all cases; however, it is computationally expensive. Among the more efficient algorithms, we find that exploiting clustering (k-means) is effective when the number of landmarks to be chosen is small (5-10). However, if 20-30 landmarks are employed, then even simple random selection works well. These results suggest that effective landmark selection appears feasible even on the scale of the Internet.

Acknowledgements

The Sockeye data used in this paper was collected at Sockeye Networks, `www.sockeye.com`. The NLANR AMP data used in this paper was collected by the National Laboratory for Applied Network Research with funding under National Science Foundation Cooperative Agreement ANI-9807479. The Skitter data was collected as part of CAIDA's Skitter initiative, `http://www.caida.org`. Support for Skitter is provided by DARPA, NSF, and CAIDA membership. The authors wish to express their appreciation to all these people and organizations for their data collection and distribution efforts.

This work was performed while Mark Crovella was at Laboratoire d'Informatique de Paris 6 (LIP6), with support from Centre National de la Recherche Scientifique (CNRS) France.

This work was partially supported by NSF grants ANI-9986397, ANI-0095988, and ANI-0322990.

References

1. The NLANR active measurement project. http://amp.nlanr.net/active/.
2. V. Athitsos and S. Sclaroff. Estimating 3D hand pose from a cluttered image. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, June 2003.
3. E. Cronin, S. Jamin, C. Jin, A. R. Kurc, D. Raz, and Y. Shavitt. Constrained mirror placement on the Internet. *IEEE Journal on sellected Areas in Communications*, 20(7):1369–1382, 2002.
4. L. Kaufman and P. J. Rousseeuw. *Finding Groups in Data: an Introduction to Cluster Analysis*. John Wiley and Sons, Inc., 1990.
5. A. Lakhina, J. W. Byers, M. Crovella, and I. Matta. On the geographic location of Internet resources. *IEEE Journal on Selected Areas in Communications, Special Issue on Internet and WWW Measurement, Mapping, and Modeling*, 2003.
6. H. Lim, J. C. Hou, and C.-H. Choi. Constructing Internet coordinate system based on delay measurement. In *Proceedings of the ACM/SIGCOMM Internet Measurement Conference (IMC-03)*, 2003.
7. E. Ng and H. Zhang. Predicting Internet network distance with coordinates-based approaches. In *Proceedings of Infocom*, 2002.
8. Y. S. P. Krishnan, Danny Raz. The cache location problem. *IEEE/ACM Transactions on Networking*, 8(5):568–582, 2000.
9. V. N. Padmanabhan and L. Subramanian. An investigation of geographic mapping techniques for Internet hosts. In *Proceedings of ACM/SIGCOMM '01*, August 2001.
10. M. Pias, J. Crowcroft, S. Wilbur, S. Bhatti, and T. Harris. Lighthouses for scalable distributed location. In *Second International Workshop on Peer-to-Peer Systems (IPTPS '03)*, Feb 2003.
11. The skitter project. http://www.caida.org/tools/measurement/skitter/.
12. Sockeye Networks. http://www.sockeye.com/.
13. L. Tang and M. Crovella. Virtual landmarks for the Internet. In *Proceedings of the ACM/SIGCOMM Internet Measurement Conference 2003*, October 2003.
14. T. Tankel and Y. Shavitt. Big-bang simulation for embedding network distances in Euclidean space. In *Proceedings of IEEE INFOCOM 2003*, April 2003.
15. J. Vleugels and R. C. Veltkamp. Efficient image retrieval through vantage objects. *Visual Information and Information Systems(VISUAL), LNCS 1614*, pages 575–584, June 1999.
16. A. Ziviani, S. Fdida, J. F. de Rezende, and O. C. M.B.Duarte. Demographic placement for Internet host location. In *Proceedings of the IEEE Global Communications Conference*, 2003.