Pharos: A Decentralized and Hierarchical Network Coordinate System for Internet Distance Prediction

Yang Chen[†], Yongqiang Xiong[‡], Xiaohui Shi[†], Beixing Deng[†], Xing Li[†]
[†]Department of Electronic Engineering, Tsinghua University, Beijing 100084, China
Email: chenyang04@mails.tsinghua.edu.cn
[‡]Microsoft Research Asia
E-mail: yqx@microsoft.com

Abstract—**Network coordinates (NC) system is an efficient mechanism for Internet distance prediction with limited measurements. In this paper, we identify the intrinsical cause for the inadequate accuracy of the simulation based NC algorithms. We then propose Pharos, a fully decentralized and hierarchical scheme, to remedy this problem. Pharos leverages multiple coordinate sets at different distance scales, with the right scale being chosen for prediction each time. We evaluate the performance of Pharos system with the King data set and latency data from PlanetLab, and compare it with the representative NC system, Vivaldi. The experimental results show that Pharos outperforms Vivaldi much without adding any significant overhead.**

I. Introduction

Network coordinates (NC) system is an efficient mechanism to predict latency between any two Internet nodes without explicit measurement to them. In NC system, each node is assigned a set of numbers called coordinates, and the network distance (latency) between any two nodes can be calculated from their coordinates with a distance function. Thus NC system significantly reduces the active probing overhead and particularly benefits large scale Internet applications, such as peer-to-peer content distribution or file sharing, multi-user gaming, and server selection.

Several approaches have been proposed in the literature, and they can be categorized into two classes in designing a NC system [3], namely landmark-based algorithm (referred as LBA in this paper) and simulation-based algorithm (referred as SBA). In LBA systems([1], [4], [5]), each node measures its distance to a set of reference nodes called landmarks, and then its coordinates can be determined by minimizing the difference between the actual distance (by measurement) and calculated distance to these landmarks. As for SBA systems, such as [2] and [13], they map nodes and pair-wise distance into a physical system, so nodes' coordinates can be determined by minimizing the energy state of the whole simulated physical system.

Recent studies have revealed that both LBA and SBA systems are short of satisfactory in term of accuracy, which is often measured by the relative error (RE) of prediction. This relative error ratio is calculated pairwisely over the whole nodes set across all distance range, while in [8] the authors found out that in a LBA system, landmarks' distribution could greatly affect the prediction accuracy, and different landmark sets should be chosen for different range of distance to improve the overall accuracy.

However, in SBA, there's no landmark in the system, the impact of the range of distance for the prediction accuracy remains unknown. Thus, in this paper, we want to answer the following questions.

1) Does range of distance of different peers affect the accuracy of prediction in SBA and how?

2) Without hierarchy of landmarks in LBA, how to design a distributed and efficient architecture for peers in SBA to remedy this problem?

The answer to the first question is positive and confirmed with a series of experiments. We analyze the distribution of the relative error of a representative SBA system, Vivaldi. In contrast to the observation for the LBA systems [8], we find out that the narrower range of distance does not lead to the higher prediction accuracy in Vivaldi, but short links still suffer from higher prediction error than long links.

As for the second problem, our main idea is to cluster the peers into different groups with different range of distance, and design one set of coordinates that is more accurate for short distances, and another set of coordinates that is more accurate for long distances. Then according to the range of distance of interest, we can choose the right set of coordinates to achieve higher prediction accuracy. Note that the number of coordinates sets can be varied depending on the scale of the range of distance. Based on this idea, we propose Pharos, a decentralized and hierarchical network coordinate system, and present its design and implementation We evaluate the performance of Pharos system with different typical data sets. Our experimental results demonstrate the improvement of the prediction accuracy comparing to the original Vivaldi system.

The rest of this paper is organized as follows. First we study the impact of range of distance and distribution on prediction accuracy for a representative SBA system, Vivaldi in Section II. Then in Section III. we present the design and implementation of Pharos, followed by its performance evaluation in Section IV. We review the related work in Section V and conclude the whole paper with Section VI.

II. Impact of Range of Distance for Vivaldi

In our work, we focus on improvement of the accuracy of Vivaldi [2], the representative of Simulation-based Algorithm.

1930-529X/07/$25.00 © 2007 IEEE
This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the IEEE GLOBECOM 2007 proceedings.

Vivaldi is the most widely used SBA system, thanks to its clean and decentralized implementation. Before describing the details, we first briefly introduce the basic procedure of Vivaldi, and then we study the causes of the prediction error of Vivaldi.

A. *Vivaldi*

Vivaldi characterizes the whole network as a spring system. Let L_{ij} be the actual distance (round-trip time) between nodes i and node j in Vivaldi system, and x_i be the coordinate assigned to node i. The coordinates of node are the result of minimizing the following error function which corresponds to the lowest energy.

$$E = \sum_i \sum_j (L_{ij} - \parallel x_i - x_j \parallel)^2 \qquad (1)$$

where $||x_i - x_j||$ is the distance between node i and node j in the chosen coordinates space.

In a decentralized Vivaldi version, each node maintains coordinates x_i and local error e_i. All nodes adjust their network coordinates and local errors through measuring their latencies to some other nodes in the system. The pseudo code for Vivaldi is shown as follows. c_e and c_c are tunable parameters.

Algorithm 1 $vivaldi(rtt, x_j, e_j)$

```
1: $w = e_i/(e_i + e_j)$
2: $e_s = | \parallel x_i - x_j \parallel - rtt \mid / rtt$
3: $e_i = e_s \times c_e \times w + e_i \times (1 - c_e \times w)$
4: $\delta = c_c \times w$
5: $x_i = x_i + \delta \times (rtt - \parallel x_i - x_j \parallel) \times u(x_i - x_j)$
```

A sample weight is firstly computed based on the local and remote error (line 1), and then the relative error is computed (line 2). Next node i updates its local error (line 3). Finally node i calculates and updates its coordinates (line 4 and line 5).

B. *Impact of range of distance for Vivaldi*

The prediction accuracy of a network coordinates scheme is often denoted by the relative error (RE) of predicted distance over the real latency measured on Internet. Relative Error (RE) is defined as [1] [2]:

$$RE = \frac{\mid PredictionDist - MeasuredDist \mid}{min(PredictionDist, MeasuredDist)} \qquad (2)$$

We use two different data sets from real Internet measurement to study the prediction error of Vivaldi. The first data set is the King data set from [2], which includes the round-trip latencies among 1740 Internet naming servers. The second data set, the PlanetLab data set, includes the round-trip latencies among 226 hosts on the Planet-Lab. This data set is downloaded from Network Coordinate Research Group at Harvard [11].

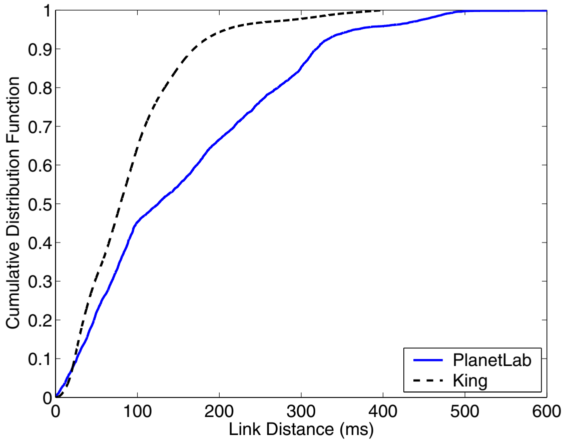


Fig. 1. Distribution of Range of Distance

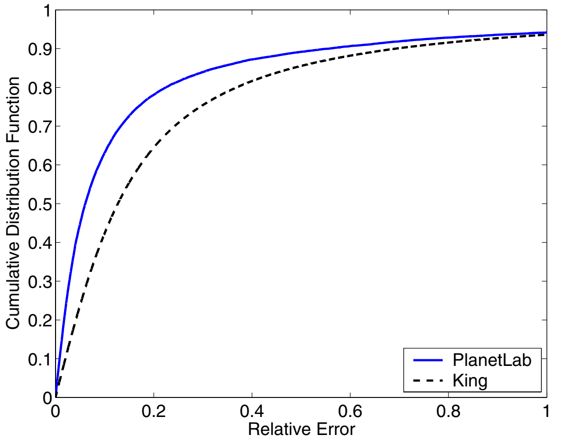


Fig. 2. Distribution of Relative Error

Fig.1 depicts the link distance distribution for these two data sets, we can see that the distance range of the PlanetLab data set is wider than that of the King data set. However, in contrast to the observation for LBA systems where wider range of distance always results in higher prediction error, in our experimental result shown in Fig.2, we find the prediction accuracy with Planetlab data set is higher than that with King data set.

We then further study how distance affects the prediction accuracy, Fig.3 depicts the distribution of the relative error over the distance spectrum. We found that the relative error varies from different ranges of distance. But, for both King data set and PlanetLab data set, we can see short links suffer from much higher relative error than long links, and we will further evaluate this in Section IV.

III. System Design

A. *Pharos Overview*

In this section, we present a new approach, called Pharos, which exploits different sets of coordinates for the same node. Each node is assigned multiple coordinates, some of which positions the node at global scale, while the others position the node at a smaller distance scale. To make it simple, we use two

1930-529X/07/$25.00 © 2007 IEEE
This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the IEEE GLOBECOM 2007 proceedings.

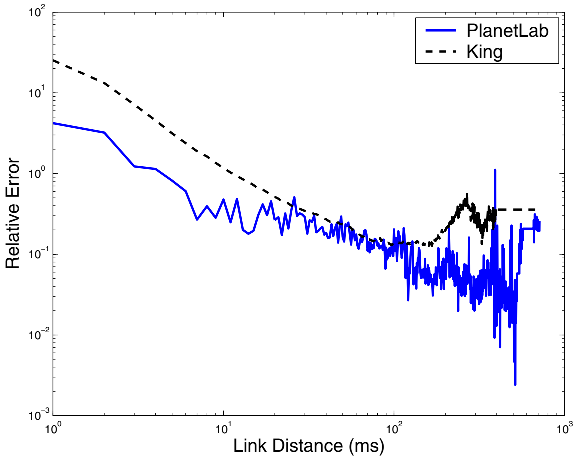


Fig. 3. Relationship between Range of Distance and Relative Error

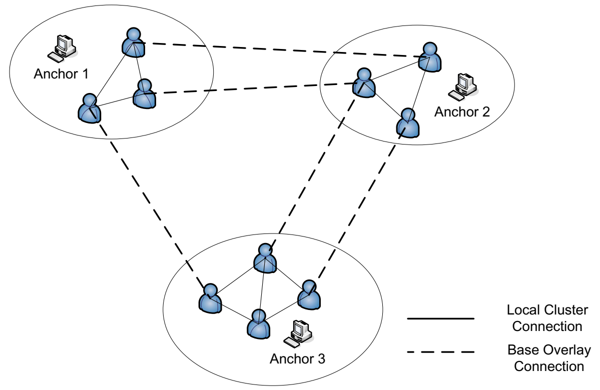


Fig. 4. Pharos Overlay

set of coordinates in Pharos, while the number of sets can be varied according to the scale of range of distance. All nodes in Pharos form two levels of overlays, namely base overlay for long link prediction, and local cluster overlay for short link prediction. And there are two types of connections: base overlay connections which are constructed randomly between nodes in the Pharos overlay; and local cluster connections which are constructed randomly between nodes in the same overlay.

In Pharos, nodes must join both base overlay and local cluster in order to have two coordinates in different prediction scale. To join the base overlay, nodes can follow the procedure presented in [2] and create k connections randomly to neighbors in the base overlay. To form the local cluster, we use the binning [12] method and choose some nodes to help node clustering called *anchors*. Any stable nodes which are able to response ICMP ping message can serve as anchor, such as the existing Internet servers (web servers or DNS servers). Guided by the anchors, nodes are grouped into different clusters as follows. Firstly a new node measures its distance to all the anchor nodes. After getting these distances, the new node can find which anchor is nearest to it and join the cluster led by the nearest anchor. Finally the joining procedure of the corresponding cluster also follows [2] and the node will create k connections randomly to neighbors in the same cluster as well. Fig.4 shows the hierarchical structure of Pharos.

Vivaldi algorithm is applied in both base overlay and local cluster. As a result, each Pharos node has two set of coordinates. The coordinates calculated in the base overlay, which we call it *global NC*, is used for the global scale, and the coordinates calculated in the corresponding local cluster, which we call it *local NC*, covers a smaller range of distance.

B. Workflow of Pharos

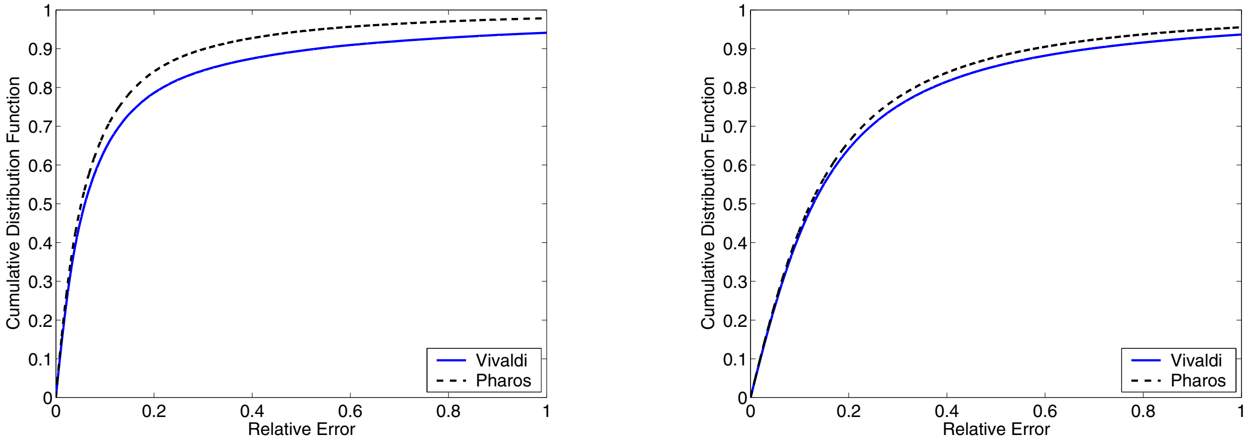
Algorithm 2 shows the procedure that a new node A joins a Pharos overlay. Node A first contacts the Rendezvous Point (RP) of the Pharos system like all other p2p schemes. After obtaining a list of anchors from RP, Host A measures the distance to all the anchors and choose the nearest cluster to join. Then node A joins both base NC overlay and local NC cluster through gossip [14] protocol. After that node A can participate in the NC calculating procedure in both base overlay and local cluster, and update their coordinates periodically.

Algorithm 2 Pharos

```
Connect_to_Rendezvous_Point(rp)
Get_Anchors_List(rp)
Nearest_Anchor_Distance = $\infty$
for $i$ in $Anchors$ do
   d(i) = Measure Distance to i
   if Nearest_Anchor_Distance $>$ d(i) then
      Nearest_Anchor_Distance = d(i)
      Nearest_Anchor = i
   end if
end for
Join_Cluster(Nearest_Anchor)
while forever do
   j = random(local neighbors of i)
   $x_{i.local} = vivaldi(rtt, x_{j.local}, e_{j.local})$
   j = random(global neighbors of i)
   $x_{i.global} = vivaldi(rtt, x_{j.global}, e_{j.global})$
   Wait(Update_Interval);
end while
```

C. Hierarchical Distance Prediction

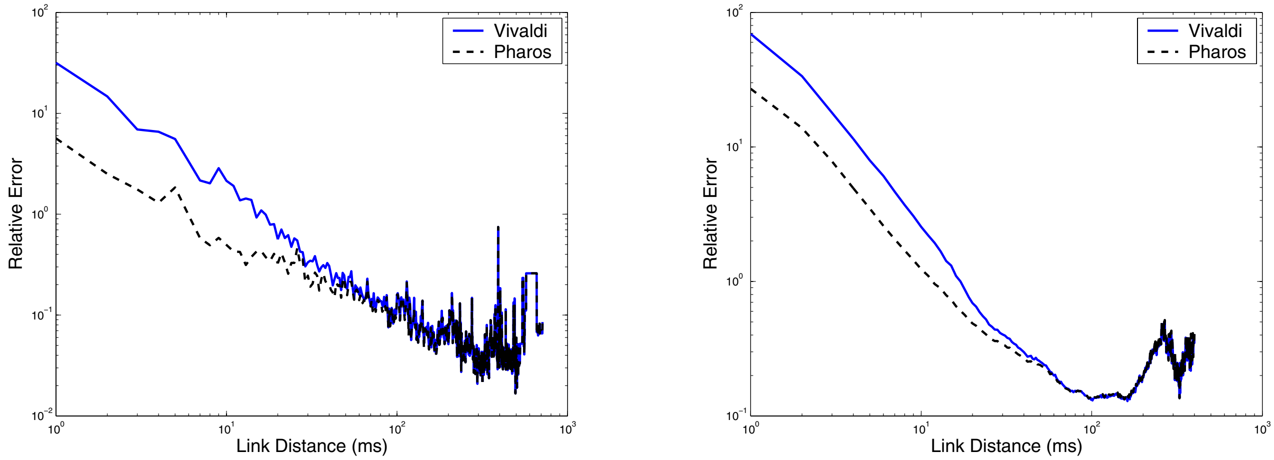
After getting the global NC and local NC, we can predict the distance between any two nodes. Distance prediction proceeds in a bottom-up fashion. If two nodes belong to the same cluster, this implies they are relatively close to each other, the distance between them is predicted by local NC. Otherwise, if these two nodes belong to two different clusters, the distance between them is predicted by global NC. This hierarchical approach would help to improve the accuracy of the distance prediction. The distance of node A and node B is defined as



(a) PlanetLab

(b) King

Fig. 5. Distribution of Relative Error



(a) PlanetLab

(b) King

Fig. 6. Relationship between Range of Distance and Relative Error

follows.

$$d = \begin{cases} \parallel x_{A.local} - x_{B.local} \parallel & cluster_A = cluster_B \\ \parallel x_{A.global} - x_{B.global} \parallel & cluster_A \neq cluster_B \end{cases} \quad (3)$$

IV. Performance Evaluation

A. Experiment Setup

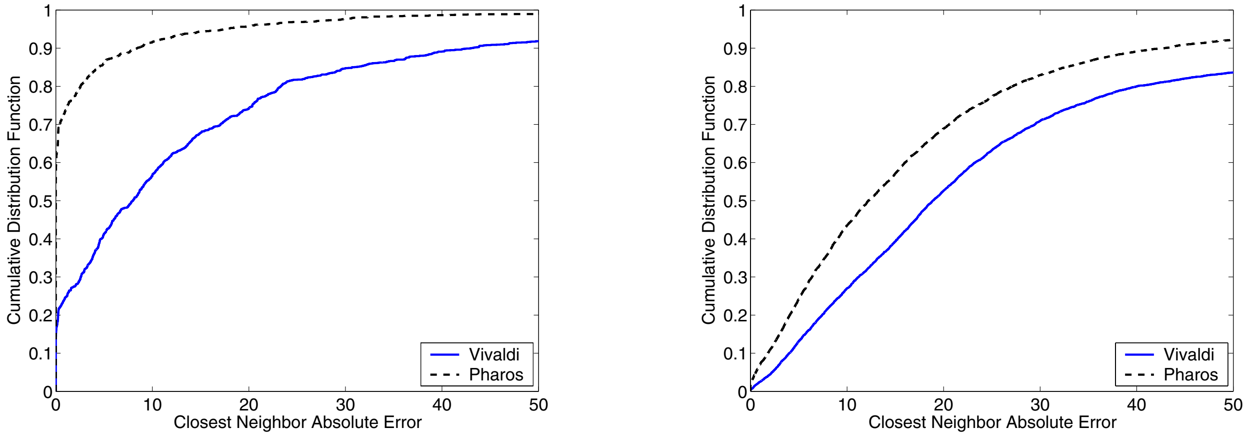
In our experiments, we compare Pharos to Vivaldi with both the King and PlanetLab data sets. Both Pharos and Vivaldi use 7-dimension coordinates. In Vivaldi, each node has 16 neighbors; Likewise, in Pharos, each node has 8 neighbors in base overlay and 8 neighbors in local cluster. Therefore, Vivaldi and Pharos have the same communication overhead. c_c and c_e in Vivaldi (also in each Vivaldi cluster in Pharos) is set to 0.25 as an empirical value in [2].

In our simulation, the nodes organize themselves into 16 proximity-based clusters. We use the k-median method [6] for node clustering and randomly choose one node from each cluster as the anchor. Ten runs are performed on each data set and the average results are reported.

B. Evaluation results of Pharos

Relative Error: Fig.5 shows the comparison of relative error between Pharos and Vivaldi. In both PlanetLab data set and King data set, Pharos outperforms Vivaldi. We pay more attention to the ninety-percentile relative error(NPRE) which would be helpful to NC-aware applications. On PlanetLab data set, the NPRE of Pharos is 0.3 while Vivaldi's is 0.52. On King data set, NPRE of Vivaldi is 0.58 and the 90 percentile relative error of Pharos is 0.69.

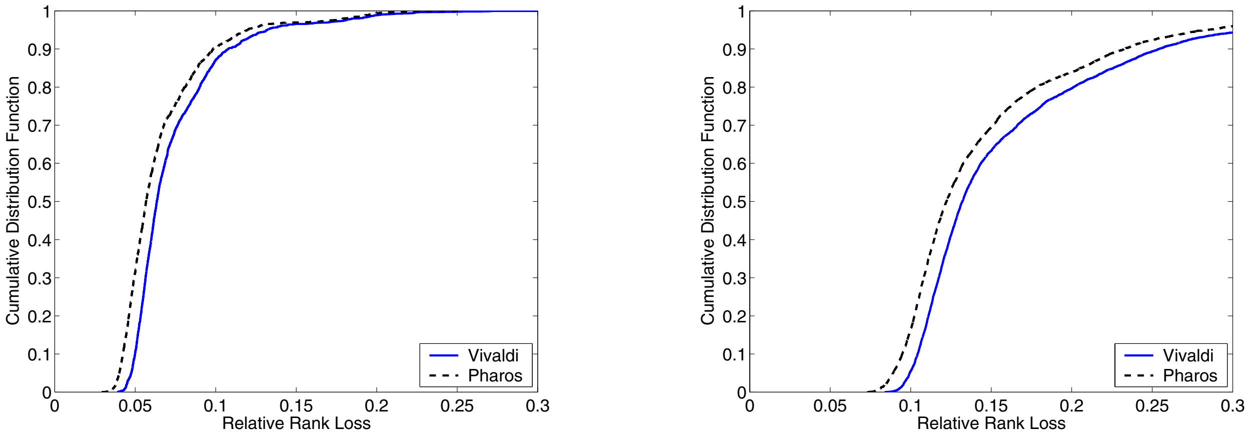
To study the impact of the distance on prediction error, Fig.6 shows the comparison of the average relative prediction error for links of various distances between Pharos and Vivaldi. Pharos improves the prediction accuracy mainly for short links while achieving almost the same prediction accuracy with Vivaldi for medium and long links.



(a) PlanetLab

(b) King

Fig. 7. Distribution of Closest Neighbor Absolute Error



(a) PlanetLab

(b) King

Fig. 8. Distribution of Relative Rank Loss

C. Other Metrics

TABLE I
Closest Neighbor Loss

Data set	Vivaldi	Pharos
PlanetLab	77.42%	30.43%
King	98.63%	95.28%

Besides relative error described in Section II, we also evaluate the performance of Pharos with the following three metrics.

Relative Rank Loss (RRL) [7] measures the probability to correctly select the closer node from an arbitrary node pair. It is defined as the percentage of incorrectly ordered node pairs (as perceived at a given node) based on the prediction.

Closest Neighbor Loss (CNL) [7] indicates the probability to correctly select the closest neighbor to a given node, and is defined by the fraction of nodes where an incorrect node is chosen as the closed neighbor using predicted distances.

In addition to the CNL, we also measure the magnitude of the error when the wrong node is selected. More precisely, we define *Closest Neighbor Absolute Error (CNAE)* as the gap between the distance to the incorrectly selected neighbor and the distance to the actual closest neighbor.

Among these metrics, relative error (RE) is the basic metric which is evaluated by all NC designers. RRL, CNL and CNAE focus more on application perspective where nodes need only to know the relative distance of other nodes.

Closest Neighbor Loss and Closest Neighbor Absolute Error: Applications will benefit from the higher prediction quality on various aspects. For example, lower CNL and CNAE, applications having higher probability to find the nearest neighbor. As illustrated in Table.I and Fig.7, in both PlanetLab data set and King data set, Pharos improves the quality of the closest neighbor selection a lot comparing to Vivaldi.

Relative Rank Loss: Fig.8 shows the comparison of relative rank loss between Pharos and Vivaldi. Similar to relative error and closest neighbor loss metrics, Pharos outperforms Vivaldi with both PlanetLab data set and King data set.

V. Related Works

Several algorithms for calculating network coordinates have been proposed. There are two classes of algorithms: landmark-based and simulation-based algorithms.

In Landmark-base algorithms (LBAs), such as GNP [1], Lighthouse [4], IDES [5], a number of nodes called landmarks are introduced to serve as reference points for other nodes to calculate their coordinates. In GNP, nodes' coordinates are computed using the Simplex Downhill method. Lighthouse derives node coordinates by solving systems of linear equations. IDES exploits matrix factorization to compute an incoming and an outgoing coordinate for each node. LBA provides high accuracy and stability, but it needs to deploy dedicated landmark nodes whose load is rather heavy to serve all the nodes in the system. This would result in single point of failure of the systems.

Simulation-based algorithms (SBAs), such as Vivaldi [2] and Big Bang Simulation [13], determine coordinates using spring-relaxation and force-field simulation, respectively. In both systems, nodes self-organize into overlay network, attract and repel each other according to network distance measurements. The low-energy state of the physical system corresponds to the coordinates with minor error. SBA systems distribute the computation and measurement to all participating nodes, so the load of each node is rather light. But joining or leaving of each node will affect the whole system, so if nodes have high churn rate, the accuracy of NC will decrease.

In [8], the authors studied the range of distance problem for landmark and explored constructing a landmark hierarchy that is shared by all nodes to improve the prediction accuracy. Specifically, a number of landmark nodes form a hierarchy through recursive clustering. Each cluster consists of landmark nodes that are close to each other.

An important difference between anchors in Pharos and landmarks in [8] is whether they are passive or active. In other words, landmarks in [8] should actively participate in the system, which means they must run NC client on landmarks for NC calculation. In contrast, the only requirement for anchors in Pharos is to reply ICMP PING request. Thus we can choose existing Internet servers such as big web servers or DNS servers as anchors in Pharos. These servers are much more stable and powerful than ordinary nodes which can improve the robustness in return. Because they only need to reply the PING query passively, we do not need to deploy Pharos client on these anchors. This makes Pharos much more practical.

Moreover, the hierarchical landmark approach in [8] needs a large number of landmark for effectively improving prediction accuracy. The number of landmarks is exponential to the number of hierarchy level. Even for a 2-level hierarchy, 256 landmark nodes are needed. As we know landmark is a critical issue in LBA systems, the deployment of large number of landmarks would become a heavy burden for the NC system designers to maintain the reliability and load balance of so many nodes in the world.

VI. Conclusion

In this paper we study the causes of the prediction error for a representative simulation based network coordination system, Vivaldi and find out that the range of distance of peers has non-trivial impact on the performance of the system. We propose a multi-set coordinates scheme called Pharos to address this issue. Our contribution is twofold. (1) We analyze the distribution of the relative error of a representative SBA system, Vivaldi and find out the relationship between range of distance of peers and the prediction error. (2) We proposes Pharos, a fully decentralized and hierarchical network coordinate system to improve the accuracy of Internet distance prediction. We evaluate Pharos system with real Internet measurement traces. The results show that Pharos achieves higher performance than Vivaldi, a representative distributed NC system. With real trace from PlanetLab network, we decrease the relative error at ninety percentile level from 0.52 (Vivaldi) to 0.3 (Pharos).

To further evaluate the practicality of Pharos, we currently focus on deploying Pharos on Internet and developing some real applications based on this NC system, such as peer-to-peer streaming and application level multicast.

Acknowledgment

This work is supported by the National Science Foundation of China (No.60473087) and National Basic Research Program of China (No.2007CB310806).

References

[1] T. S. E. Ng and H. Zhang. Predicting Internet Network Distance with Coordinates-Based Approaches. In Proc. of INFOCOM, June 2002.

[2] F. Dabek, R. Cox, F. Kaashoek, and R. Morris. Vivaldi: A Decentralized Network Coordinate System. In Proc. of SIGCOMM, August 2004.

[3] J. Ledlie, P. Gardner, and M. Seltzer. Network Coordinates in the Wild. In Proceedings of NSDI, April 2007.

[4] M. Pias, J. Crowcroft, S. Wilbur, et al. Lighthouses for Scalable Distributed Location. In Proc. of IPTPS, February 2003.

[5] Y. Mao and L. K. Saul. Modeling Distance in Large-Scale Networks by Matrix Factorization. In Proc. of IMC, October 2004.

[6] K. P. Gummadi, S. Saroiu, and S. D. Gribble. King: Estimating Latency between Arbitrary Internet End Hosts. In Proc. of SIGCOMM IMW, November 2002.

[7] E. K. Lua, T. Griffin, M. Pias, et al. On the Accuracy of Embeddings for Internet Coordinate Systems. In Proc. of IMC, October 2005.

[8] R. Zhang, Y. C. Hu, X. Lin, et al. A Hierarchical Approach to Internet Distance Prediction. In Proc. of ICDCS, 2006.

[9] L., Kaufman, P., J., Rousseeuw. Finding Groups in Data: An Introduction to Cluster Analysis. John Wiley and Sons, New York. (1990).

[10] PlanetLab, http://www.planet-lab.org/.

[11] NC Research Group at Harvard,http://www.eecs.harvard.edu/ syrah/nc/.

[12] S. Ratnasamy, M. Handley, R. Karp, et al. Topologically-Aware Overlay Construction and Server Selection. In Proc. of INFOCOM, June 2002.

[13] Y. Shavitt and T. Tankel. Big-Bang Simulation for Embedding Network Distances in Euclidean Space. In Proc. of INFOCOM, April 2003.

[14] Y. H. Chu, A. Ganjam, T.S.E. Ng, et al. Early deployment experience with an overlay based Internet broadcasting system, in USENIX Annual Technical Conference, Jun. 2004.