Polar : An Efficient Finding Nearest Neighbor Algorithm for Overlay Network

Tan Chen , Xiangzhi Sheng and Baosong Shan
State Key Laboratory of Software Development Environment
School of Computer Science and Engineering
Beijing University of Aeronautics and Astronautics
Beijing , China
{chentan,xzsheng,shanbs}@nlsde.buaa.edu.cn

Work In Progress

Abstract

Obviously, network proximity is an essential characteristic for overlay network construction. By selecting the nearest neighbor for the target node, large-scale distributed applications can decrease the communication cost and improve the performance significantly. Nevertheless, existing mechanisms for exploiting network proximity information are either inaccurate or low efficiency. In this paper, we present a novel and practical algorithm to find the nearest neighbor in overlay network, called Polar. Performing direct measurement based on the result of distance prediction using network coordinate, Polar achieves accuracy as while as it has low overhead. Meanwhile, our proposal does not rely on the static landmark so it has strong scalability and high robustness. Polar is a generic architecture that can be used for a variety of overlay network with different requirement.

Categories and Subject Descriptors

C.2.4 [**Computer-Communication Networks**]: Distributed Systems – *Distributed applications*.

General Terms

Algorithm , Design , Performance

Keywords

overlay network , nearest neighbor , network coordinate

1. INTRODUCTION

Over the past several years, considerable works have been done in the area of building topology aware overlay network. Briefly, there are two methods on the whole: expanding ring search and network coordinate. Expanding ring search[1] has to blindly flood a large number of nodes to obtain a reasonable result. Consequently, the overhead is the main obstacle to employ this kind of method in real environment. Network coordinate[2] predicts the proximity of nodes in the overlay through embedding network distance measurements in a coordinate system. Nevertheless, network embedding approach is neither accurate nor complete and complex mechanisms besides virtual coordinate are required to support NC in large-scale distributed applications.

In this paper, we propose Polar, a novel and practical algorithm to find the nearest neighbor in overlay network which combines distance prediction using network coordinate and round-trip time measurements to generate proximity information, achieving both efficiency and accuracy. Meanwhile, Polar does not rely on the static landmark so it has high scalability and robustness. This algorithm is a generic architecture that can be used for a variety of overlay network with different requirement.

2. DESIGN AND IMPLEMENTATION

2.1 Data Structure

The design of Polar is fully distributed, combining the network coordinate and direct measurement to achieve accuracy as while as degrade the overhead. In order to achieve efficient query, the main problem is how one Polar node selects and organizes other nodes it knows, called neighbors, in the overlay network. Each Polar node maintains a data structure which puts the set of neighbors into concentric, non-overlapping rings, and the range of the ith ring is $[as^{i-1}, as^{i}]$, where a is a constant and s is the multiplicative increase factor . By all appearances, the innermost ring, which is much special than others in Polar, is from 0 to a, while the outermost ring spans to ∞ .

In 2 dimensional Euclidean space, by drawing a series of equiangular lines from each node, these rings, except the innermost ring, are divided into some pie slices. Each pie slice represents a special region in Internet, which contains the neighbors whose characteristic is similar in terms of Internet latency. In another words, these neighbors are close each other. Each node predicts the distance to its neighbor, calculates the angle between the line containing the two nodes and the x axis, and places the neighbor in the corresponding pie slice. These parameters represent a tradeoff between accuracy and overhead.

Intuitively, with the help of this data structure, neighbors are geographically dispersive, and each neighbor can respond the geographic queries for its region of the network quickly and accurately. The exponentially increasing ring radii improves the querying performance. Like [3], this data structure can resolve queries in O(logN) steps. The node preserves abundant and

INFOSCALE 2007, June 6-8, Suzhou, China

DOI 10.4108/infoscale.2007.953

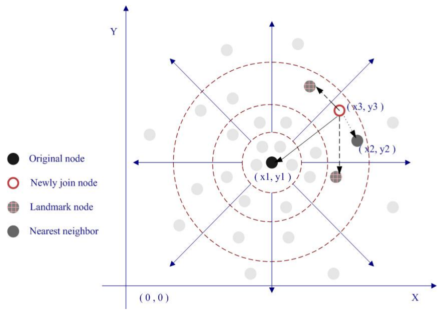

Figure1. Data Structure and Search Procedure.

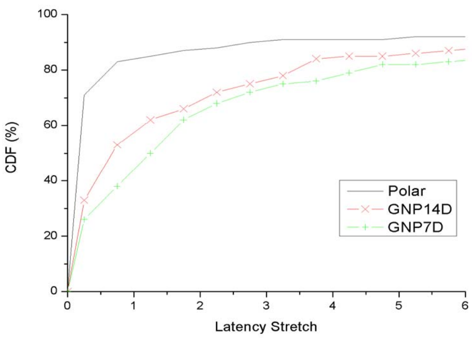

Figure2. CDF of Latency Stretch for King data.

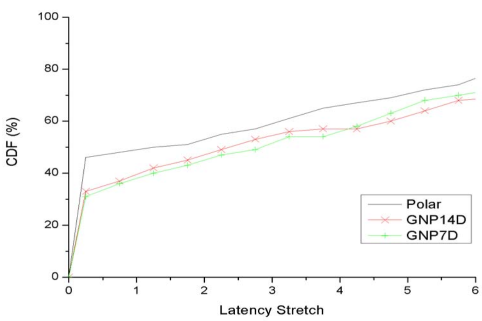

Figure3. CDF of Latency Stretch of TS data.

exact information of its neighbors in the immediate regions, as well as retains a sufficient number of neighbors in remote regions. As a result, a query may be routed to the specialize region the target node belongs to efficiently.

2.2 Search Procedure

When a new node wants to join the network, it needs to know at least one node in the system as the bootstrapping node, whom it should contact at first. With the latency l measured between these two nodes, the bootstrapping node randomly chooses two neighbors in the ring, whose R_i and r_i satisfied with $r_i < l < R_i$. These three nodes act as the landmarks for the newly joining node, and utilizing simplex downhill algorithm, new node gets its coarse-gain coordinate and the query is routed to the closet one of the landmarks.

The procedure continues until that node gets into the innermost ring of a node be requested in the sequence. At that time, the candidate pool is small enough to perform direct measurement and the newly joining node will find its nearest neighbor in the overlay network. Fig.1 illustrates this process. The newly joining node contacts the requested node whose coordinate is (x1,y1) firstly. Based on the distance between them, the requested node selects two landmarks for the new one. With the help of these three nodes, the new node will get its coordinate in this space and find its nearest neighbor.

3. EVALUATION

We tested our polar system on both Internet measurement data set and simulated topology. The latency datasets we used are as follows: The first one is from the P2PSim[4] project. It contains the pair-wise RTTs between 1709 Internet DNS servers. The second one is based on Transit-Stub[6] topology with 1,000 nodes.

The metric used to evaluate the algorithms is latency stretch, defined as :

$$\frac{\text{predicted shortest distance} - \text{measured shortest distance}}{\text{measured shortest distance}}$$

We compared Polar with GNP[5]. We configured GNP for 14 dimensions and 7 dimensions. For Polar, we use 5 rings per node and 8 pie slices per ring , the radii of inner-most ring is 15ms. 100 targets from two dataset respectively were selected randomly. Fig.3 and Fig.4 show the cumulative distribution function(CDF) of the latency stretch . As we can see from the figures, Polar can perform significantly better than GNP. 50-60% of the nodes correctly select the closest neighbor.

4. CONCLUSION

This paper describes Polar, an efficient algorithm to find nearest neighbor in overlay network. Polar achieves both efficiency and accuracy because it performs direct RTT measurement based on the result of distance prediction using network coordinate. Polar is also robust and scalable because it uses dynamic landmarks and not rely on central server to distributed information. Therefore, Polar can be used to improve the performance of many large-scale distributed applications like network-aware overlay construction and location of nearby resources in the network.

5. REFERENCES

[1] M. Waldvogel and R. Rinaldi. Efficient Topology-Aware Overlay Network. *HotNets 2002, SIGCOMM/CCR 2003*

[2] Peter R. Pietzuch, Jonathan Ledlie, Michael Mitzenmacher, Margo I. Seltzer. Network-Aware Overlays with Network Coordinates. *ICDCS Workshops* 2006: 12

[3] B. Wong, A. Slivkins, and E. G. Sirer. Meridian: A Lightweight Network Location Service without Virtual Coordinates. In *Proc. of SIGCOMM'05*, Aug. 2005.

[4] The P2PSim Project. http://pdos.csail.mit.edu/p2psim/ .

[5] T. S. E. Ng and H. Zhang. Predicting Internet Network Distance with Coordinates-Based Approaches. In *Proc. of INFOCOM'02*, June 2002.

[6] E. Zegura, K. Calvert, and S. Bhattacharjee. How to model an Internetwork. *IEEE INFOCOM'96*, San Francisco, CA, May 1996.