Improving Chord Network Performance Using Geographic Coordinates

Sonia Gaied Fantar[1] and Habib Youssef[2]

Research Unit Prince, ISITC Hammam Sousse, University of Sousse, Tunisia
soniagaied3@gmail.com
habib.youssef@fsm.rnu.tn

Abstract. Structured peer-to-peer overlay networks such as Chord, CAN, Tapestry, and Pastry, operate as distributed hash tables (DHTs). However, since every node is assigned a unique identifier in the basic design of DHT (randomly hashed), "locality-awareness" is not inherent due to the topology mismatching between the P2P overlay network and the physical underlying network. In this paper, we propose to incorporate physical locality into a Chord system. To potentially benefit from some level of knowledge about the relative proximity between peers, a network positioning model is necessary for capturing physical location information of network nodes. Thus, we incorporate GNP (Global Network Positioning) into Chord (Chord-GNP) since peers can easily maintain geometric coordinates that characterize their locations in the Internet. Next, we identify and explore three factors affecting Chord-GNP performance: distance between peers, message timeout calculation and lookup latency. The measured results show that Chord-GNP efficiently locates the nearest available node providing a locality property. In addition, both the number of the messages necessary to maintain routing information and the time taken to retrieve data in Chord-GNP is less than that in Chord.

1 Introduction

A large number of structured Peer-To-Peer overlay systems constructed on top of DHT such as Chord [1], CAN [2], Tapestry [4] or Pastry [3] have been proposed recently. Due to their scalability, robustness and self-organizing nature, these systems provide a very promising platform for a range of large-scale and distributed applications.

In the structured P2P model, the nodes in the network, called peers, form an application-level overlay network over the physical network. This means that the overlay organizes the peers in a network in a logical way so that each peer connects to the overlay network just through its neighbors. However, the mechanism of peers randomly choosing logical neighbors without any knowledge about underlying physical topology can cause a serious topology mismatching between the P2P overlay network and the physical underlying network. The topology mismatching problem brings a great stress in the Internet infrastructure and greatly

N. Trigoni, A. Markham, and S. Nawaz (Eds.): GSN 2009, LNCS 5659, pp. 77–86, 2009.

limits the performance gain from various lookup or routing techniques. Meanwhile, due to the inefficient overlay topology and the absence of relationship between the node's location and the node's identifier, the DHT-based lookup mechanisms cause a large volume of unnecessary traffic. Moreover, due to this discrepancy, DHTs do not offer a guarantee on the number of physical hops taken during a lookup process, as a single overlay hop is likely to involve multiple physical routing hops. Aiming at alleviating the mismatching problem and reducing the unnecessary traffic on Chord, we propose a novel location-aware identifier assignment function. This function attributes identifiers to nodes by choosing physically closer nodes as logical neighbors. Thus, the main contribution of this paper is the design and analysis of a new approach that incorporates locality-awareness into Chord identifiers, by assigning identifiers to nodes that reflects their geographic disposition. So, we propose to use a coordinates-based mechanism, called Global Network Positioning (GNP), to predict Internet network distance. Chord-GNP is constructed on the basis of Chord aiming to achieve better routing efficiency, which attributes peer's identifiers by choosing physically closer nodes as logical neighbors. The main optimizations in Chord-GNP are lower overlay hops and lookup latency.

The remainder of this paper is organized as follows.

We discuss related work in Section 2. In Section 3, we present Chord in detail. Section 4 talks about the Global Network Positioning (GNP). In sections 5 and 6 we analyze our contribution and give simulation results. Section 7 concludes this paper and gives a brief outlook on our future work.

2 Related Work

Many efforts have been made to improve locality awareness in decentralized structured peer-to-peer overlays. The most widely used approaches in locality awareness are network proximity. Castro et al [5] divide techniques to exploit network proximity into three categories: expanding-ring search, heuristics, and landmark clustering. The entries of routing table are chosen as the topologically nearest among all nodes with node's identifier in the desired portion of the key space [6]. The success of this technique depends on the degree of freedom an overlay protocol has in choosing routing table entries without affecting the expected number of routing hops. Another limitation of this technique is that it does not work for overlay protocols like CAN and Chord, which require that routing table entries refer to specific points in the key space.

3 CHORD

Chord is a peer-to-peer protocol which presents a new approach to the problem of efficient location. Chord [1] uses consistent hashing [7] to assign keys to its peers. Consistent hashing is designed to let peers enter and leave the network with minimal interruption. This decentralized scheme tends to balance the load on the system, since each peer receives roughly the same number of keys, and there is little movement of keys when peers join and leave the system. In a steady state, for N peers in the system, each peer maintains routing state information for about only $O(logN)$ other peers (N number of peers in the system). The consistent hash functions assign peers and data keys an m-bit identifier using SHA-1 [8] as the base hash function. A peer's identifier is chosen by hashing the peer's IP address, while a key identifier is produced by hashing the data key. The length of the identifier m must be large enough to make the probability of keys hashing to the same identifier negligible. Identifiers are ordered on an identifier circle modulo $2m$. Key k is assigned to the first peer whose identifier is equal to or follows k in the identifier space. This peer is called the successor peer of key k, denoted by successor(k). If identifiers are represented as a circle of numbers from 0 to 2^{m-1}, then successor(k) is the first peer clockwise from k. The identifier circle is termed as the Chord ring.

To maintain consistent hashing mapping when a peer n joins the network, certain keys previously assigned to n's successor now need to be reassigned to n.

4 Predicting Internet Network Distances with Coordinates Based Approaches

Among several categories of approaches that predict internet network distance, the coordinates based approaches may be the most promising. Several approaches have been proposed among which GNP [9][10] may have received the most attention.

4.1 Global Network Positioning (GNP)

GNP [9][10] is a two-part architecture that is proposed to enable the scalable computation of geometric host coordinates in the Internet. In the first part, a small distributed set of hosts called Landmarks first compute their own coordinates in a chosen geometric space. The Landmarks' coordinates serve as a frame of reference and are disseminated to any host who wants to participate. In the second part, equipped with the Landmarks' coordinates, any end host can compute its own coordinates relative to those of the Landmarks.

Landmark Operations. The first part of the architecture is to use a small distributed set of hosts known as Landmarks to provide a set of reference coordinates necessary to orient other hosts. When a re-computation of Landmarks' coordinates is needed over time, we can ensure the coordinates are not drastically changed if we simply input the old coordinates instead of random numbers as the start state of the minimization problem. Once the Landmarks' coordinates are computed, they are disseminated to any ordinary host that wants to participate in GNP.

Ordinary Host Operations. In the second part of our architecture, ordinary hosts are required to actively participate. Using the coordinates of the Landmarks in a geometric space, each ordinary host now derives its own coordinates.

To do so, an ordinary host H measures its round-trip times to the N Landmarks using ICMP ping messages and takes the minimum of several measurements for each path as the distance. Using the measured host-to-Landmark distances, host H can compute its own coordinates that minimize the overall error between the measured and the computed host-to-Landmark distances.

5 Proposition

One important issue in alleviating the mismatching problem and reducing the unnecessary traffic on Chord is to attribute peer's identifiers by choosing physically closer nodes as logical neighbors. Thus, we can benefit from predicting internet network distances. Specifically, we propose to use a coordinates-based approaches for network distance prediction in Chord architecture. The main idea is to ask peers to maintain coordinates (i.e. a set of numbers) that characterize their locations in the Internet such that network distances can be predicted by evaluating a distance function over hosts' coordinates. Our contribution is to allow participating peers in Chord to collaboratively construct an overlay based on physical location. And, to potentially benefit from some level of knowledge about the relative proximity between the peers, the Global network Positioning approach (GNP) [9][10] is integrated into Chord for capturing physical location information of network peers. This paper presents a topology-based node identifier assignment for Chord that attempts to map the overlay's logical key space onto the physical network such that neighboring nodes in the key space are close in the physical network.

5.1 Descriptions

Using the Chord lookup protocol, the peers are assumed to be distributed uniformly at random on the ring. In particular, there is a base hash function which maps peers, based on their IP adresses, to points on the circle, i.e. that it maps identifiers to essentially random locations on the ring (Figure 1.a). However, in Chord [1], when a node joins the network, it is not optimally positioned in the ring in respect of the underlying network such as IP network. For this purpose, we propose a novel location-aware identifier assignation function for Chord. This function attributes identifiers to nodes by choosing physically closer nodes as logical neighbors by using a coordinates-based mechanism (Figure 1.b). Chord could potentially benefit from some level of knowledge about the relative proximity between its participating nodes by using the coordinates based approaches GNP to predict internet network distance : Chord-GNP.

5.2 Location-Aware Identifier Assignation Function for Chord

In this section, we describe a design technique whose primary goal is to reduce the latency of Chord routing. Not unintentionally, this technique offers the additional advantage of improved Chord robustness in term of routing.

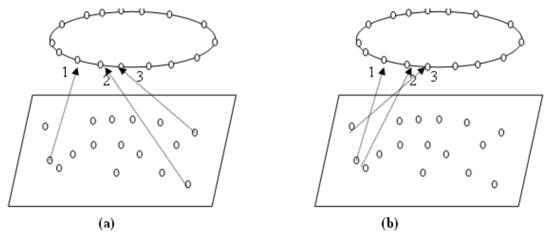

Fig. 1. (a) mapping nodes to identifiers with consistent hashing. (b) mapping nodes to identifiers with locality aware.

In our representation, we model the Internet as a particular geometric space S. Let us denote the coordinates of a host H in S as C_H^S. Then, each peer in the Internet is characterized by its position in the geometric space with a set of geometric coordinates C_H^S.

We want to replace the Chord's base hash function SHA-1 (Node identifier = SHA-1(IP address, port number)) by the location-aware identifier assignation function that generates geometric coordinate's identifiers: GNP(IP address, port number)=C_H^S

$$id_N = C_N^S \tag{1}$$

5.3 Design

Node joining and leaving in Chord-GNP is handled like the Chord does. The difference is that in Chord, a node's identifier is chosen by hashing the node's IP address, but Chord-GNP provides location-aware identifier assignation function. This function assign peer's identifier designating its position in the virtual ring. We will describe the three most basic pieces of our design: Chord-GNP routing, construction of the Chord-GNP coordinate overlay, and maintenance of the Chord-GNP overlay.

Routing in Chord-GNP. Chord-GNP also uses a one-dimensional circular key space. Chord-GNP's main modification to Chord is to include new identifiers into Chord's routing tables, i.e. Chord-GNP inherits Chord's successor list and finger table to use in Chord-GNP's routing algorithm and maintenance algorithm. Each node has a successor list of nodes that immediately follow it in the coordinate's space.

Each node keeps a list of successors: If a node's successor fails, it advances through this list until a live node is found. Routing efficiency is achieved with the finger list of nodes spaced exponentially around the coordinate's space.

Intuitively, routing in Chord-GNP network works by following the finger table through the ring from source to destination coordinates.

Chord-GNP construction. To allow the Chord-GNP to grow incrementally, a new node that joins the system must derives its own coordinates that characterize its location in the Internet to be allocated in the ring. This is done by the coordinates based approaches: GNP. In GNP, the Internet is modeled as a D-dimensional geometric space. Peers maintain absolute coordinates in this geometric space to characterize their locations on the Internet. Network distances are predicted by evaluating a distance function over peers' coordinates.

A small distributed set of peers known as Landmarks provide a set of reference coordinates. peers measure their latencies to a fixed set of Landmark nodes in order to compute their coordinates. While the absolute coordinates provide a scalable mechanism to exchange location information in a peer-to-peer environment, the GNP scheme presented in Section 4 used distance measurements to a fixed set of Landmarks to build the geometric model. When a node joins the Chord-GNP network it will be placed in the ring by choosing physically closer nodes as logical neighbors. The successor pointers of some nodes will have to change. It is important that the successor pointers are up to date at any time because the correctness of lookups is not guaranteed otherwise. The Chord-GNP protocol uses a stabilization protocol running periodically in the background to update the successor pointers and the entries in the finger table.

Node departure, recovery and Chord-GNP maintenance. To leave an established Chord-GNP ring, a node can give its keys to its successor and then inform its predecessor. The successor and predecessor nodes then update their fingers tables and successors lists. Chord-GNP ensures also that each node's successor list is up to date. It does this using a "stabilization" protocol that each node runs periodically in the background and which updates Chord's finger tables and successor pointers. In Chord-GNP, when the successor node does not respond or fails, the node simply contacts the next node on its successor list.

6 Simulation Results

In this section, we evaluate the performance of Chord-GNP through simulation. The goal of this evaluation is to validate the proposition of Section 5. That proposition assumed idealized models of Chord.

We are implementing optimizations of GNP coordinates in the current open source implementation of the Chord distributed hash table as described in [1] Overlay Weaver [11][12]. We modified the Chord simulator [11] to implement GNP.

6.1 Distance between Peers

In an experiment, we first bring up a network of four nodes placed in 2 classrooms (figure 2). 1) Chord: Figure 3 shows screenshots of simulation which visualizes communication between nodes just in time. Nodes are sorted on the ring on the basis of their ID, taking into account the Hash algorithm as identifier assignation function. Each node has to maintain a virtual link to its successor, which is the node directly following it in the ordered node set. With this structure, any node can route messages to any other node simply by each intermediate node forwarding the message to its successor until the destination is reached. This, however,

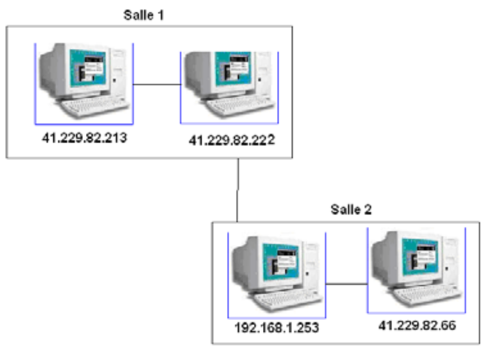

Fig. 2. Arrangement of Peers

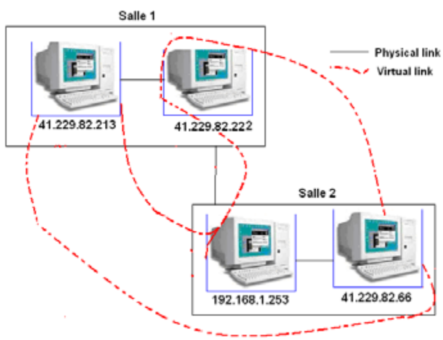

Fig. 3. Identifier assignation function is Hash algorithm SHA-1

results in choosing of the next peers, rendering routing not scalable. As was mentioned in Figure 5, node 41.229.82.222 routes messages to node 41.229.82.213 by intermediate node 41.229.82.66, not knowing that node 41.229.82.213 is closer to node 41.229.82.222.
2) Chord-GNP: In this section we report the results of testing Chord using identifiers provided by GNP.
In this experiment, Nodes are sorted on the ring on the basis of their coordinates given by GNP (Figure 4).
With this structure, node 41.229.82.222 can route messages directly to its successor which is 41.229.82.213.

6.2 Message Timeout Calculation

The performance penalty of routing in the overlay over taking the shortest path in the underlying network is quantified by the message timeout calculation. Chord sends messages periodically to maintain overlays. On a real network, we conducted experiments with four computers. We invoked a DHT shell on each

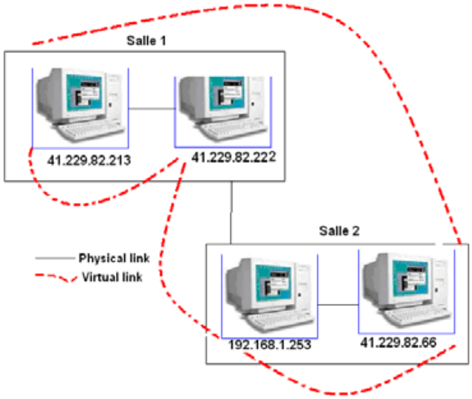

Fig. 4. Identifier assignation function is GNP

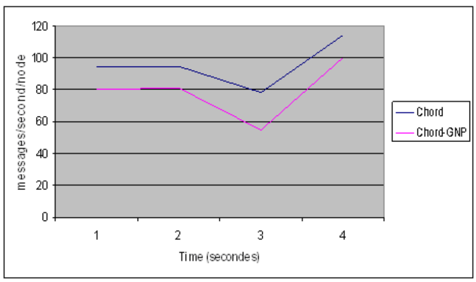

Fig. 5. Number of messages per second per node being passed between four real computers

computer and controlled them via a network. We counted the numbers using the message counter. The control scenario was as follows: after 4 nodes had joined an overlay, a node put a value on a DHT every 2 seconds 4 times. Figure 5 plot the average number of messages per second per node for Chord and Chord-GNP. We claim that the number of messages per second per node being passed between Chord-GNP's nodes are less those sent over Chord's nodes. These experiments show that Chord-GNP reduces the number of messages for Chord. In fact, the reducing of the number of messages is improved by the choice of physically closer nodes as logical neighbors.

6.3 End-to-End Path Latency

This section presents latency measurements obtained from implementations of Chord and Chord-GNP. We produce several lookup scenarios with the number of nodes increased from 10 to 300 nodes. Then, we generate a simulation topology for these scenarios and evaluate the end-to-end path lookup latencies. Figure 6

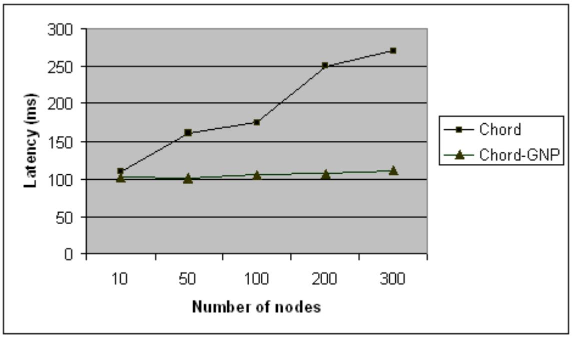

Fig. 6. Lookup latency with the number of nodes increased

shows that only where the number of nodes is small, the latency in both systems is the same. In addition, the lookup latency of Chord increases as the number of nodes is increased, while the latency of Chord-GNP remains almost at the same low level. It is an important advantage of Chord-GNP that its performance is independent of the number of participating nodes, which confirms our prediction. Consequently, the performance of Chord-GNP becomes much better than that of Chord.

7 Conclusions

In this paper, we have studied a new class of solutions to the Internet distance prediction problem that is based on end hosts-maintained coordinates, called Global Network Positioning (GNP). We have proposed to apply this solution in the context of a peer-to-peer architecture, precisely in the Chord Algorithm. This topology-based node identifier assignment attempts to map the overlay's logical key space onto the physical network such that neighboring nodes in the key space are close in the physical network. Chord-GNP's main routing optimizations are of less overlay hops and passing proximity links of the underlay network. Meanwhile, Chord-GNP has insignificant lookup latency in comparison to Chord.

References

1. Stoica, I., Morris, R., Karger, D., Kaashock, M., Balakrishman, H.: Chord: A scalable P2P lookup protocol for Internet applications. In: Proc. of ACM SIGCOMM (2001)
2. Ratnasamy, S., Francis, P., Handley, M., Karp, R., Shenker, S.: A scalable content addressable network. In: Proc. of ACM SIGCOMM (August 2001)
3. Rowstron, A., Druschel, P.: Pastry: Scalable, decentralized object location and routing for large-scale p2p systems. In: Proc. of IFIP/ACM Middleware (2001)
4. Zhao, B., Huang, L., Stribling, J., Rhea, S.C., Joseph, A., Kubiatowicz, J.: Tapestry: A global-scale overlay for rapid service deployment. IEEE J-SAC 22(1) (2004)

5. Castro, M., Druschel, P., Hu, Y.C., Rowstron, A.: Exploiting Network Proximity in Peer-to-Peer Overlay Networks. In: International Workshop on Future Directions in Distributed Computing (FuDiCo), Bertinoro, Italy (June 2002)
6. Hong, F., Li, M., Yu, J., Wang, Y.: PChord: Improvement on Chord to Achieve Better Routing Efficiency by Exploiting Proximity. In: ICDCS Workshops 2005, pp. 806–811 (2005)
7. Karger, D., Lehman, E., Leighton, T., Panigrahy, R., Levine, M., Lewin, D.: Consistent hashing and random trees: distributed caching protocols for relieving hot spots on the world wide web. In: Proceedings of the twenty-ninth annual ACM symposium on Theory of computing, May 1997, pp. 654–663 (1997)
8. Secure hash standard, NIST, U.S. Dept. of Commerce, National Technical Information Service FIPS 180-1 (April 1995)
9. Ng, T.S.E., Zhang, H.: Predicting internet network distance with coordinates-based approaches. In: Proceedings of IEEE Infocom (May 2002)
10. Ng, T.S.E., Zhang, H.: Towards Global Network Positioning. In: ACM SIGCOMM Internet Measurement Workshop, San Francisco, CA (November 2001)
11. Overlay Weaver: an overlay construction toolkit, `http://overlayweaver.sf.net/`
12. Shudo, K., Tanaka, Y., Sekiguchi, S.: Overlay Weaver: An overlay construction toolkit Computer Communications (Special Issue on Foundations of Peer-to-Peer Computing) 31(2), 402–412 (2008) (available online on August 14, 2007)