Solving Overlay Mismatching of Unstructured P2P Networks using Physical Locality Information

Jia Zhao[1] Jiande Lu[2]
Soochow University, Jiangsu Suzhou, 215006, P.R.China
[1]Jason.soochow@gmail.com, [2]lujiande@public1.sz.js.cn

Abstract *This paper addresses Overlay Topology Mismatching problem by presenting "AM", an effective way to measure nodes' relative distance and a technique called "CTAG" to construct topology-aware Gnutella overlay. The approach excels in sending no extra messages to probe the target nodes, avoiding the extra traffic and delay time, while keeping the necessary accuracy.*

1. Introduction

According to related studies [1], P2P traffic has contributed the largest portion of Internet traffic, among which a large amount of traffic is caused by Overlay Topology Mismatching [2]. This problem is common to many overlays, as the solutions do not take into account the difference between the overlay topology and the underlying physical topology. This problem is most obvious in unstructured P2P overlays such as Gnutella and KazAa, which, at the same time, may be the most commonly used P2P applications on the nowadays Internet. This kind of situation may be the results of Gnutella bootstrapping process [3], in which peers blindly choose their overlay neighbors from node list returned by random GWCs (Gnutella Web Cache).

There are two ways to solve the mismatching problem. (1) Constructing Overlay using physical topology information at first hand. (2) Finding the close neighbors when communicating. Our work addresses these two phases. Besides, AM and CTAG also serve as two of the enabling techniques for work [4][5].

2. Critical Topology-Aware Grouping (CTAG)

Specifically presented, CTAG (Critical Topology-Aware Grouping) groups the close nodes in Gnutella network into different local groups in order to support the low-cost and low delay communications between nodes. It works both for constructing the overlay and dynamically revising the overlay during the nodes' interaction.

2.1 Grouping Strategy Foundation

As is known that IP address assignment was solely managed by IANA (Internet Assigned Numbers Authority) before 1999 and the IP address was allocated according to the need that was uneven. Later, five RIRs (Regional Internet Registry) have been setup to distribute the work by managing assignment of their blocks of IP addresses. Thus, the assignment strategies of IANA and respective RIRs provide the foundation of the CTAG: the adjacent nodes within the same organization are always assigned with IP addresses within the same block. For the simplicity of demonstration, this paper uses 5 regions for P2P networks as the corresponding RIR region that shown in Table 1.

2.2 Adjacency Measurement (AM)

Based on section 2.1, we can calculate the "IP distance" between two nodes to denote the relative physical distance. More specifically, the calculation criterion is the longest matched IP segment: the algorithm starts from the highest segments of the 2 nodes' IP addresses. If the 2 segments match, AM increases by 1. The algorithm stops once the segments does not match. Then, AM could be 3, 2, 1 or 0, which denote the 2 nodes are in the same region with corresponding matched segments. If AM=0 or 1, we need to refer to Table 1 to determine the actual AM. (1) When AM=0, the 2 nodes may or may not reside in the same region. (2) When AM=1, 2 nodes are possible in the different regions. Thus AM could be 1 or –1 (2 nodes are in different region). Note that mixed groups will affect the accuracy since IP address from this group may come from more than 1 region, as they were "Direct Allocated" before 1999. For example, if there are following nodes: A (202.195.128.45), B (202.195.128.175), C (202.123.43.5), D (58.12.42.231), E (65.213.42.142), therefore: $AM_{AB} = 3$, $AM_{AC} = -1$, $AM_{AD} = 0$, $AM_{AE} = -1$, $AM_{BC} = -1$, $AM_{BD} = 0$, $AM_{BE} = -1$, $AM_{CD} = -1$, $AM_{DE} = -1$.

2.3 Two Phases Grouping in Gnutella

This section demonstrates the CTAG's two phases grouping process in Gnutella network by simply modifying the existing protocols.

2.3.1 Bootstrapping Grouping – Phase 1

When bootstrapping, Gnutella nodes use following steps [7] to try to get node list and connect to the network: **1.**Launch the client and load the host cache. **2.**Try to connect to the GNet, using the host cache. **3.**If after X seconds, there's no connection to the GNet, query a random

Table 1 IP address Blocks Assignment Table [6]

RIR	IP Blocks		
APNIC	058/6, 122/6, 169.208.0.0/12, 218/6, 202/7 *, 121/8, 126/8, 196.192.0.0/13, 210/7, 222/8 43/8, 133/8		
ARIN	24/8, 63/8 - 76/8, 199/8, 204/8 - 209/8, 216/8, 3/8, 4/8, 6/8, 8/8, 9/8, 11/8-13/8, 15/8-22/8, 26/8, 28/8-30/8, 33/8-35/8, 38/8, 40/8, 44/8-50/8, 52/8, 54/8-56/8, 128/8-132/8, 135/8-137/8, 140/8, 142/8, 143/8, 152/8, 154/8, 155/8, 160/8, 170/8, 171/8, 214/8, 215/8		
RIPE	62/8, 80/8 - 91/8, 193/8 - 195/8, 212/8 - 213/8, 217/8, 25/8, 51/8, 53/8, 57/8, 145/8		
LACNIC	189/8 - 190/8, 200/8 - 201/8	AfriNIC	41/8, 202.123.0.0/19(* this block is transferred from APNIC to AfriNIC)
Mixed	134/8, 138/8, 139/8, 141/8, 146/8-151/8, 153/8, 156/8-159/8, 161/8-169/8, 172/8, 188/8, 191/8, 192/8, 196/8, 198/8		

Note: Blue color IP blocks denote Direct Allocated ones before 1999. We divide them into corresponding groups. Mixed ones include IP blocks from more than 1 region.

Proceedings of the Sixth IEEE International Conference on Peer-to-Peer Computing (P2P'06)
0-7695-2679-9/06 $20.00 © 2006 IEEE

GWC. **4.**Wait for at least Y seconds before sending a request to another GWC. **5.**After Y seconds, if no results have been received from the first GWC call, call a new GWC each Z seconds, until a response is received from one of them.

At the first launch, Gnutella node will go to step 3 to find connectable nodes to join the network. Especially in Limewire, there are 181 GWC servers hard coded to assist the bootstrapping process [3]. We first modify step 3, let nodes find the closest default GWC server to get node list by referring $AM_{node,GWC}$. Second, we modify step 5 to call another closest GWC every Z seconds when no reply from the previous GWC call instead of calling randomly. If all AM value are the same (e.g. AM=2), or if all AM=-1, which means there is no available GWC server in the same region with the current node, then call a random GWC Note that all nodes need to maintain some remote connections in avoid of "Island" effect caused by CTAG (Figure1, gray links in c, d, e, f).

2.3.2 Dynamic Revision – Phase 2

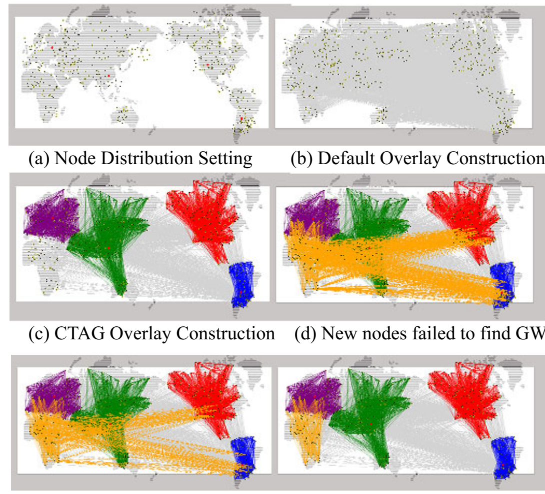
During the operation, Gnutella nodes can get the node list through following ways [7]: **1.**Calling a GwebCache. **2.**Storing hosts' addresses read from X-Try headers during handshakes. **3.**Storing hosts' addresses in pong messages. **4.**Storing hosts' addresses read from QueryHit messages

As in the phase 1, we modify step 2 to 4 by referring to AM value to guide the node when establishing connections with the closest nodes. If the node reaches the max neighbor connections, node should disconnect the ones with the lowest AM.

3. Experiment & Result Analysis

We have tested the CTAG in our parallel simulation system called HIVE(Hybrid Interlinked Versatile Environment). We simulate a P2P system with 500 nodes with randomly generated IP address to simulate the assignment. We define 5 regions, 4 of which have pre-installed GWC servers to assist the bootstrapping. The results are shown in Figure 1. First, by adopting phase 1, nodes within the same region can find each other via calling the closest GWC when first entering the network (c, color links), because the closest GWC is more possible to return nodes that are close to it when all nodes adopt this strategy. While using default approach, nodes will randomly choose the neighbors that may not close to them physically (b, gray links). This will cause much more unnecessary long-range traffics and delay the communication between nodes. Second, we simulate the situation that nodes cannot find a GWC in their region and hence have to register to GWCs in other regions (section 2.3.1). This will cause these nodes connect to physically remote nodes (d, orange links). Nevertheless, by adopting phase 2, these nodes will no longer blindly choose neighbors, but by referring to AM value, they will gradually find more and more physically close nodes through the interaction. As shown in (e) and (f), nodes in region 5 finally connect to the close neighbors while still maintain some remote connections as all other nodes in region 1 to 4.

This new topology-aware overlay provides the foundation for selective query in Gnutella and will significantly cut the redundant traffics and reduce the query delay between the



(a) Node Distribution Setting (b) Default Overlay Construction

(c) CTAG Overlay Construction (d) New nodes failed to find GWC

(d) Dynamic Revision 3 times (f) Dynamic Revision 11 times

Local connection for region 1-5

Remote connection/Default Overlay connection

Figure1 Simulation Results

nodes. We compared the communication cost between settings as in Figure 1(b) and 1(c) in a continuous running process in order to get a steady result. As shown in Figure 2, new overlay bears much less cost than the original one. Besides, since the simplicity of CTAG, it could be used to help other overlays' construction. There shall be no conflicts with other techniques. The defect is the accuracy of AM. Due to the lack of order in early allocation, the IP addresses in mixed group might be misplaced. The result accuracy depends on the required system precision.

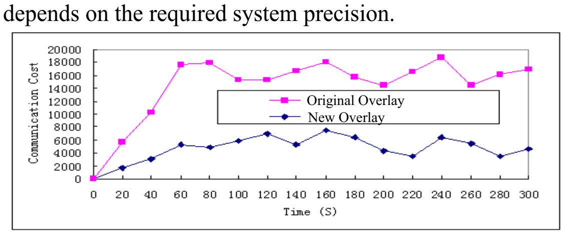


Figure 2 Communication Cost Comparison

4. Conclusions

AM presented in the paper excels in that it does not send out extra messages to probe the target nodes. Therefore, it does not cause extra traffic and delay time, while keeping the necessary accuracy. Based on AM, CTAG is able to construct a topology-aware overlay for Gnutella or other similar unstructured overlays, providing the foundation to support the selective query.

Reference

[1]. S. Saroiu, K. P.Gummadi, R. J. Dunn, S. D. Gribble, and H. M. Levy, An Analysis of Internet Content Delivery Systems, SOSDI2002.

[2]. Yunhao Liu, Xiao Li, Lionel M. Ni, Yunhuai Liu, Overlay Topology Matching in P2P Systems, GCC 2003, Shanhai, China

[3]. Pradnya Karbhari, Mostafa Ammar, Amogh Dhamdhere, Himanshu Raj, George Riley, and Ellen Zegura Bootstrapping in Gnutella: A Measurement Study, PAM 2004, LNCS 3015, pp. 22－32, 2004.

[4]. Jia Zhao, Jiande Lu, Pyramid: Building Incentive Architecture for Unstructured Peer-to-Peer Network, ICIW 2006, France, Feb. 2006

[5]. Jia Zhao, Jiande Lu, Search Optimized and Self-Organized P2P Community Management Framework, CNAIS05, China, pp. 593-597, 2005

[6]. IPv4 Address Space, http://www.iana.org/assignments/ipv4-address-space

[7]. Gnutella Specification, http://rfc-gnutella.sourceforge.net