Heuristic Relay Node Selection Algorithm for One-hop Overlay Routing

Yin Chen[1], Li Tang[1,2], Jun Li[2,3]
[1] Department of Automation, Tsinghua University, Beijing, China
[2] Research Institute of Information Technology, Tsinghua University, Beijing, China
[3] Tsinghua National Lab for Information Science and Technology, Beijing, China
{chenyin03@mails., tangli03@mails., junl@}tsinghua.edu.cn

***Abstract*—This paper reviews the characteristics of overlay networks and defines effective relay nodes that can improve the performance of interactive real-time applications. A heuristic relay node selection algorithm for overlay routing is proposed. The algorithm can find effective relay nodes for end-to-end links in an overlay network, and is scalable and robust due to its decentralization and randomization. Based on the simulation results, the algorithm is able to maintain a good performance when 40% end nodes in an overlay network encounter failures.**

***Index Terms*—Overlay networks, relay routing, relay node selecting**

I. Introduction

Internet makes access to and exchange of information much more convenient than ever before. Its 'best-effort service' model ensures robustness and scalability on system level, and prospers numerous applications such as web, ftp and email. However, emerging applications are putting some novel performance demands on the Internet, such as bandwidth, packet loss rate, end-to-end (E2E) latency, availability, etc. Those requirements are not guaranteed by the best-effort service. Many researches have demonstrated that the E2E performance of the Internet is far from optimal [8, 9, 10].

Overlay network is an effective way to support new applications and protocols without changing the Internet's underlying network infrastructure. In an overlay network, the hosts are logically connected, where an overlay path may consist of single or multiple IP-layer hops. As the IP layer has already provided generic connectivity between every pair of overlay nodes, overlay network may give participating nodes the additional flexibility to select paths for application specific objectives, which is called the overlay routing. A number of works have shown that overlay routing is capable of providing much better E2E performance and resilience on the Internet [1, 2, 3, 4, 5, 6, 7, 14].

In this paper, we propose a novel relay node selection algorithm for overlay routing, named as Heuristic One-hop Relay Node Selection (HORNS), which specifically aims to provide overlay routes for interactive applications such as VoIP. Decentralized and randomized, HORNS is suitable to select effective relay nodes for all E2E links in an overlay network. Due to its inherent robustness, it is able to maintain a performance even when up to 40% hosts in an overlay network encounter failures.

The rest of this paper is organized as follows. Section II introduces related work and points out the differences between HORNS and previous approaches, and then states the goals of HORNS. Section III briefly describes the test data used in simulation experiments. Section IV analyzes the benefits of one-hop overlay routing and gives the definition of effective relay nodes. Section V presents HORNS in detail, and Section VI validates its effectiveness with various simulations. Finally, Section VII concludes the paper.

II. Related Work

Overlay network has been a hot research spot for years. Resilient overlay networks (RON) [1] uses overlay routing to bypass IP-layer path failures. It monitors the availability of IP-layer paths between every pair of participating nodes, and uses overlay paths to forward data packet when the direct paths fail. Due to its full mesh architecture, RON is not scalable over 50 overlay nodes. One-hop Source Routing (OHSR) [3] randomly picks k candidate relay nodes and chooses the best one to form a one-hop overlay path. Work in [4] leverages multiple overlay paths to reduce packet loss rate. In [7], VoIP data packets are cached in a ring-buffer on each relay node in order to retransmit the missing packets from the relay node nearest to the destination rather than from the source. PPRR [5] maintains a pool of candidate relay nodes for each destination node so that the source node can pick relay nodes from this pool when the direct path to the destination node is unavailable.

RON and other similar systems need to probe the full mesh networks and are hence not scalable. Although OHSR is scalable due to its randomized manner, it may potentially eliminate good relay nodes. Instead of targeting the whole networks, PPRR only focuses on each destination node. HORNS differs from previous work in that it leverages decentralization and randomization to provide a pool of candidate relay nodes for each source node, out of which effective relay nodes can be picked for most destination nodes.

As the first of design goals, we require the relay node selection algorithm to be robust and scalable. Therefore, the proposed HORNS differs from RON in that it does not

monitor the status of all IP-layer paths between every pair of nodes, and instead each node only keeps tracking a small fraction of overlay links. Similar to OHSR that randomly chooses candidate relay nodes, HORNS also adopts the randomness manner, but HORNS introduces additional heuristic information that significantly improves the quality of the selected relay nodes. Unlike PPRR, HORNS maintains a pool of candidate relay nodes for each participating node. In other words, each node in the overlay network keeps a relay node pool; when a node need to communicate with another one but the direct IP layer path fails or cannot satisfy the desired performance request at that time, at a high possibility, it is able to successfully find a useful relay node from its pool and then use it to forward data packets with satisfied E2E performance.

As mentioned, HORNS is designed to provide service specifically for interactive applications; hence we use round trip time (RTT) as performance metric of paths because most interactive applications are contingent upon E2E delay.

In summary, HORNS is a randomized and decentralized relay selection algorithm for one-hop overlay routing.

III. Data Set

In this section we will give the detailed description of our experiments. We use the RTT data collected from PlanetLab [15] and build a virtual network test bed in p2psim [16]. PlanetLab is a global-scale distributed research platform including hundreds of end nodes all over the world. P2psim is a discrete event simulator for comparing, evaluating, and exploring peer-to-peer protocols.

Specifically, the RTT data are collected by the all-pairs-pings (APP) project [17]. APP deploys a set of scripts on each PlanetLab node and asks the nodes to periodically ping each other at a 15 minutes interval and record the corresponding RTT measurements. Based on these data, we generate a network topology for p2psim. As the RTT is the only parameter of this topology, it is therefore the only metric of E2E performance. This is reasonable for most real-time interactive applications where the quality of service is mainly contingent upon E2E delay.

We use the data collected on Sep 22, 2005, where the network conditions were relatively stable [12]. The APP data of this day contain 487 end nodes. Therefore, our virtual network test bed consists of 487 end nodes and 236682 one-way IP-layer E2E paths.

To calculate the RTT of a one-hop overlay path, we add the source-node-to-relay-node RTT to the relay-node-to-destination-node RTT, without considering the transmission time on the relay node. Previous work indicates that relaying packets is a CPU insensitive task and the transmission time is only a few milliseconds [13], and is thus trivial compared with the several hundred milliseconds RTT.

IV. Benefits of One-hop Overlay Routing

The details of HORNS will be given in Section V. This section will show to what extent the improvement can be gained via one-hop overlay routing. This determines the upper bound on the amount of improvement that a sub-optimal approach can achieve.

We find the best relay node for each source-destination-node pair, and compare the RTT of the overlay path with its corresponding IP-layer E2E path. As shown in Table I, there are 236682 IP-layer E2E paths in our virtual overlay network, including 84072 outage paths. Among the outage paths, around 14% can be repaired via one-hop relay nodes.

About 83% of the available IP-layer paths can achieve lower RTT through one-hop relay routing. Fig. 1 shows the CDF of the RTT reduction ratios of these 127024 one-hop overlay paths with regard to their direct IP-layer paths. The RTT reduction ratio is defined as (direct RTT – relay route RTT) / (direct RTT). We can see about 50% of the optimal one-hop overlay paths reduce the E2E RTT by less than 10%. Nevertheless, there exist 10% overlay paths that can reduce IP-layer E2E RTT by as much as 50%.

The International Telecommunication Union (ITU) G.114 standard [11] suggests 150ms to be the upper limit of one-way delay for most satisfied real-time interactive applications, which means the upper limit of RTT for those applications should be no more than 300ms. Therefore, we define that an ***effective relay node*** for a pair of end nodes is such a node that if it is chosen as the relay node for the node-pair, the E2E RTT of this one-hop overlay path is no greater than 300ms. Although this cannot guarantee that the one-way delay is below 150ms due to the asymmetry of IP routing, it is acceptable in practice.

According to Table II, 41% of IP-layer E2E paths with RTT above 300ms can reduce the RTT below 300ms through one-hop relay routing. The average RTT of these direct paths and their one-hop relay routes are 915ms and 208ms respectively. Among the outage IP-layer paths, 11.4% can get RTT no greater than 300ms by using relay nodes.

The algorithm proposed in this paper aims to select effective relay nodes instead of optimal ones. The optimal relay nodes refer to those nodes that can provide the shortest one-hop overlay path in terms of RTT among all nodes in the overlay network. Finding the optimal relay nodes is unnecessary. For one thing, all effective relay nodes are sufficient for interactive applications; for another, finding the optimal relay node incurs larger overheads than finding an effective relay node.

V. Description of HORNS

In Fig. 3, node U and V stand for two end nodes in the overlay network. U maintains a candidate relay node pool denoted by N in its local memory. When U detects that the direct IP-layer path between U and V is congested or cannot satisfy the application's performance requirement, U will pick a node W from N and use W to forward data packets to V, and vise versa. When U needs to communicate with another end node Y, and the direct IP-layer path between U and Y is broken or cannot meet the application specific requirements, U also will try to select a relay node from the same set N. An end

TABLE I BENEFITS OF OVERLAY ROUTING

Direct path	Number of links	Number of improved links	Ratio	Average RTT of relay paths
All	236682	138856	59%	224 ms
Outage	84072	11832	14%	294 ms
Working	152610	127024	83%	218 ms

TABLE II BENEFITS OF OVERLAY ROUTING IN REGARD WITH THE 300MS THRESHOLD

Direct path type	Number of links	Number of relay links with RTT <= 300 ms	Ratio	Average RTT of direct paths	Average RTT of relay paths
Working paths with RTT above 300ms	32813	13404	41%	915 ms	208 ms
Outage paths	84072	9601	11.4%	NA	151 ms

node maintains one and only one set N.

In summary, HORNS works in a decentralized manner: each end node in the overlay network maintains an set N; every single end node has its own set N that can be used to provide relay nodes when necessary.

The major challenge of HORNS is how to select nodes to be put into set N. Unlike OHSR [3] that randomly picks *k* nodes to form a candidate pool, HORNS leverages some effective heuristic information. Here we show this heuristic information. Fig. 2 depicts the CDF of E2E RTT of direct IP-layer paths and the RTT between source nodes and relay nodes in the corresponding optimal one-hop relay paths. As shown in Fig. 2, if randomly choosing nodes to form N, then the distribution of RTT between U and intermediate relay nodes in N will follow the solid line, which is different from the dash-dot line that shows the distribution of RTT between U and the optimal relay nodes. Therefore, we design HORNS to maintain set N in a way that the distributions of RTT between source nodes and nodes in set N are similar to the dash-dot line.

The size of set N is a small integer, currently 20. We will explain why the number 20 is selected in section VI. Now, let's assume the size of set N is 20. Because of this small size, the distributions of RTT between source nodes and nodes in set N cannot be exactly the same to the dash-dot line. Alternatively, we observe that as shown by the dash-dot line, about 50% RTT are in the [0, 50]ms region, and about 20% RTT are in the [50, 100] ms region, and so on. Therefore, HORNS assures that 50% nodes in N have the RTT between them and the source node within 50ms, 20% between 50ms and 100ms, etc. Within each region, nodes are randomly selected. Therefore, HORNS is partially randomized with some restrictions.

In fact, for each source-destination pair there exist a certain

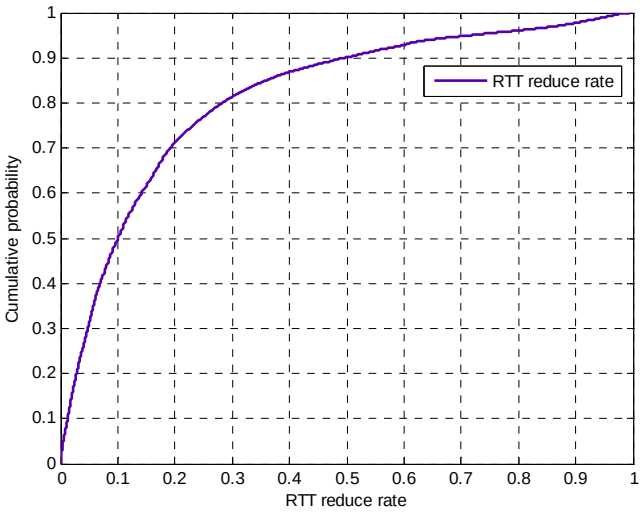

Fig. 1 CDF of RTT reduce rate by leveraging one-hop relay routing

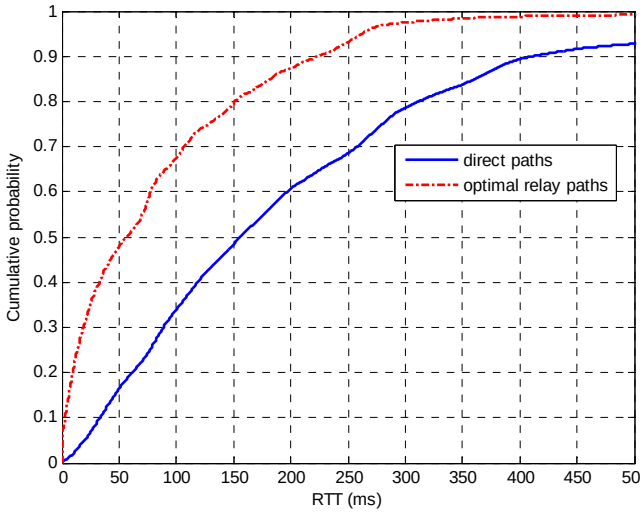

Fig. 2 CDF of end-to-end RTT of direct paths and source-to-relay RTT between source nodes and optimal relay nodes of optimal relay paths

quantity (might be zero) of effective relay nodes. Intuitively, we can consider that these nodes compose a set. Since there are multiple source-destination pairs, there are also multiple sets. Combining all these sets, we can get a large set denoted by M. The key idea of HORNS is to maintain a set N that tends to imitate the set M. Inspired by the difference between the solid line and dash-dot line in Fig. 2, we suppose that the M set is different from the whole set denoted by Q. As the overlay network expands, Q set will grow and the difference between M and Q should be more significant. This is conceptually straightforward: if Q is small and the RTT between any pair of end nodes is less than 50ms, then every node is an effective relay node, hence the M set and Q set are identical; in contrast, if Q set is so large that a considerable fraction of E2E RTT are beyond 300ms, then the M set differs a lot from the Q set. According to the difference between M and Q, we expect that

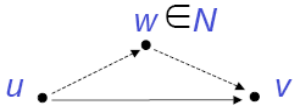

Fig. 3 Model of HORNS

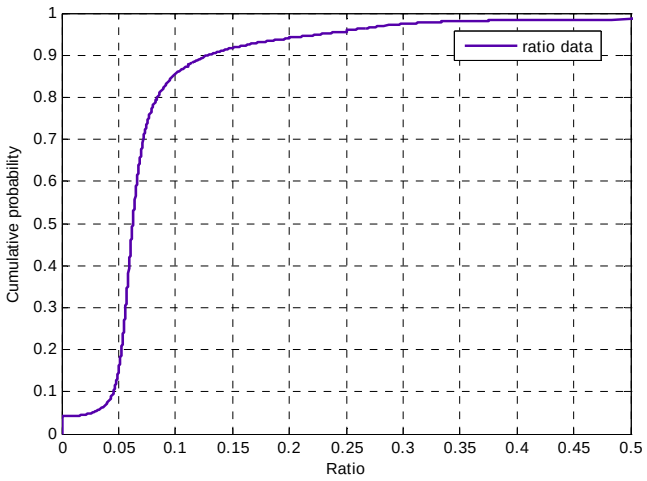

Fig. 4 CDF of the ratio of the number of effective relay nodes in set N to the total number of all effective relay nodes within the overlay network

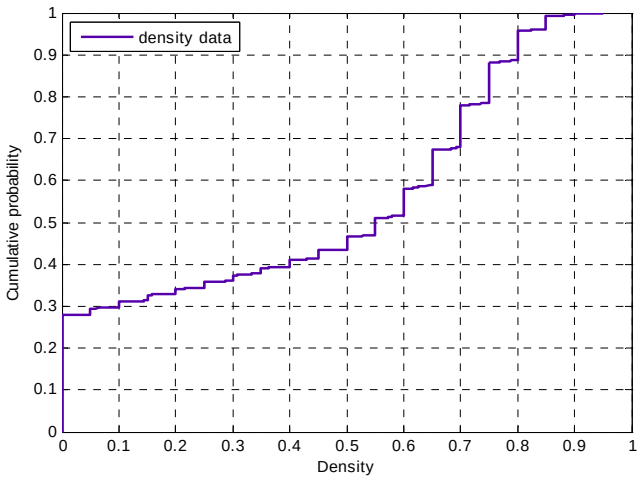

Fig. 5 CDF of the density of effective relay nodes in set N

when Q grows, the density of effective nodes in the whole network becomes scarcer, while the density of effective nodes in the M set should stay relatively much higher.

Note that the density of effective nodes in the M set is not 100%. This is because by definition as long as a node is an effective relay node for at least one source-destination node-pair, it would be contained in the M set. Therefore, a node in the M set is not always effective for any given pair of source-destination nodes.

Since N tends to ensemble M, the density of effective nodes in N should be consistent with M. As a consequence, it should be worthwhile to find effective relay nodes from N, which is the fundamental assumption of HORNS. Simulation results in Section VI validate its effectiveness.

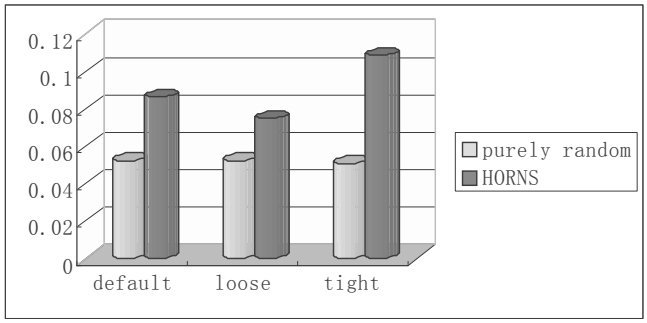

Fig. 6 Comparison of purely random algorithm and HORNS

Currently, set N in HORNS has 20 elements, which means each node monitors 20 relay nodes. As mentioned before, the reason that this number is chosen will be explained in section VI. Now, suppose the node U is running HORNS. At the start of every round, node U queries the nodes in its set N to fetch their set N. Then it put all these nodes into a temporary P set. Afterwards it pings all the nodes in the P set to get the RTT measurements. Then U partitions the RTT into several sections and randomly selects nodes from each section at a given quantity so that the distribution of RTT between the U and the nodes in its set N ensembles the dash-dot line in Fig. 2. During the execution, HORNS also adds the nodes that have communicated with U to its P set, to make sure that the algorithm will not be dead locked even if the P set becomes empty once in a while. After one round of execution, HORNS will wait a given length of time before the start of next execution.

VI. Simulation Results and Analysis

As a decentralized algorithm, HORNS does not intend to find 100% effective nodes. On the other hand, set N contains only 20 candidates of intermediate nodes; therefore, a source node can find at most 20 effective relay nodes for each destination node. As a consequence, set N inherently can only provide a fraction of all effective nodes that exist in the whole overlay network. Fig. 4 shows the CDF of this fraction. According to Fig. 4, if there exists effective relay nodes for an IP-layer E2E path in the whole overlay network, the probability for HORNS to find at least one effective relay node is above 95%. This clearly shows the motivation behind the selection of the size of set N. With a size 20, set N can give a 95% success ratio.

Dividing the number of effective relay nodes in a set N with the size of set N, we can get the density of effective relay nodes in set N. Fig. 5 plots the CDF of the density. As can be seen, for more than 70% sessions, the corresponding set N can provide at least $0.1 \times 20 = 2$ effective relay nodes, which is a reasonable figure and further validates our selection of the size of set N.

To compare HORNS with the purely random algorithm, we hereby give two more specific definitions of effective relay nodes in addition to the default one given in Section IV:

a) *A loose definition*: an effective relay node for a pair of

TABLE III NUMBER OF NODES ALIVE IN SET N WHEN THE NETWORK EXPERIENCES NODE OUTAGES

MTTR	Number of outage nodes	Average number of nodes alive in each set N
2	100	14.7951
2	200	11.3036
2	300	8.2647
2	400	5.3649
5	100	15.4437
5	200	12.322
5	300	9.9191
5	400	7.6604

end nodes is such a node that if the pair of end nodes chooses it as the relay node, the E2E RTT of this one-hop relay route is either no greater than 300ms ***or*** less than the direct path.

b) *A tight definition*: an effective relay node for a pair of end nodes is such a node that if the pair of end nodes chooses it as the relay node, the E2E RTT of this one-hop relay route is both no greater than 300ms ***and*** less than the direct path.

Under the loose definition, the total number of effective nodes in the overlay network is higher than in the default definition; whereas under the tight definition, the total number of effective nodes in the overlay network is lower than in the default definition.

Fig. 6 compares the average ratios of the effective relay nodes in set N to the total number of all effective relay nodes in the whole overlay network. The horizontal axis denotes three types of definitions of effective relay nodes; the vertical axis denotes the ratio. It can be seen that as the definition of effective relay nodes turns tighter, the advantage of HORNS over purely random algorithm becomes more significant. This is because when the definition is tighter, there are fewer effective relay nodes within the whole network, which makes it more difficult for the purely random algorithm to find effective relay nodes. Thanks to the heuristic information, HORNS wins over the purely random algorithm. The nature of the aforementioned One-hop Source Routing (OHSR) [3] is actually purely random, therefore the test results also means that HORNS wins over OHSR.

As the overlay network scales, the effective relay nodes are certainly to become scarcer, in which case HORNS is expected to show its superiority to the random algorithm according to Fig. 6. The total number of end nodes in the data set used in this paper cannot be expanded. Alternatively, we draw Fig. 6 by tuning the number of effective nodes through their definitions.

Robustness is another important design goal of HORNS. Due to its randomization and decentralization, HORNS is expected to be highly robust. Theoretically, since the nodes in set N are randomly selected, it is reasonable to anticipate that even if 50% nodes experience outages and those nodes are distributed evenly in the whole overlay network, the proportion of alive nodes in set N should be also around 50%.

To justify this, we conduct experiments with eight groups of parameters. First, we let the number of outage nodes to be 100, 200, 300 and 400; then in each case, we set the mean time to repair (MTTR) of each outage node as 2 and 5 algorithm loops (defined in Section V) respectively. The time to repair (TTR), denoting the duration of an outage, is a negative-exponentially-distributed random variable whose expectation is MTTR. The TTRs of different nodes are identical and independent from each other. With these parameters, we let HORNS run for 100 algorithm loops and count the number of active nodes in every set N.

As we can see in Table III, the average number of active relay nodes decreases when the number of outage nodes grows. When the MTTR decreases, the outage incidents occur more frequently, leading to the drop of average number of live nodes. Since our overlay network consists of 487 nodes and the size of set N is 20, the second and sixth rows show that even when up to 40% nodes fail, the fraction of active nodes in set N can still stay above 50%, making the survival end nodes still able to receive satisfactory service with HORNS.

VII. CONCLUSION

In this paper we proposed a one-hop relay node selection algorithm, named HORNS. HORNS works in a randomized and decentralized manner, and is robust against node failures.

For each source node, HORNS provides a candidate relay nodes pool to provide relay nodes for any given destination nodes. In order to improve its density of effective relay nodes, HORNS constructs the candidate pool by selecting nodes based on the RTT distribution heuristic represented in the set of optimal relay nodes, rather than randomly choosing nodes from all the participating end nodes. Simulation results show that HORNS outperforms the purely random relay selection algorithm and One-hop Source Routing (OHSR), and is robust against node failures.

ACKNOWLEDGMENT

This work is sponsored by NEC Laboratories China. The authors are grateful to the developers of APP project for sharing the data, and would like to thank all the members of FIT Security Lab for their discussion and advices.

REFERENCES

[1] D. G. Andersen, N. Feamster, S. Bauer, and H. Balakrishnan. Resilient overlay networks. In Proceedings of the 18th ACM Symposium on Operating Systems Principles, 2002.

[2] S. Savage, T. Anderson, A. Aggarwal, D. Becker, N. Cardwell, A. Collins, E. Hoffman, J. Snell, A. Vahdat, G. Voelker, and J. Zahorjan. Detour: a case for informed Internet routing and transport. IEEE Micro. 1999, 19 (1): 50-59.

[3] K. P. Gummadi, H. V. Madhyastha, S. D. Gribble, H. M. Levy, and D. Wetherall, "Improving the Reliability of Internet Paths with One-hop Source Routing," In Proceedings of 6th Symposium on Operating System Design and Implementation (OSDI), 2004.

[4] D. G. Andersen, A. C. Snoeren, and H. Balakrishnan. Best-path vs. multi-path overlay routing. In Proceedings of the 3rd ACM SIGCOMM Conference on Internet Measurement, 2003.

[5] C. Cheng, Y. Huan, H. Kung, and C. Wu. Path probing relay routing for achieving high end-to-end performance. In Proceedings of Global Telecommunications Conference, 2004.

[6] S. Ren, L. Guo, and X. Zhang. ASAP: an AS-aware peer-relay protocol for high quality VoIP. In Proceedings of IEEE ICDCS'06, 2006.
[7] Y. Amir, C. Danilov, S. Goose, D. Hedqvist, and A. Terzis. 1-800-overlays: Using overlay networks to improve voip quality. In NOSSDAV, 2005.
[8] V. Paxson, End-to-end routing behavior in the Internet, IEEE/ACM Transactions on Networking, vol. 5, no. 5, pp. 601~615, 1997.
[9] S. Neil, M. Ratul, and A. Thomas, Quantifying the causes of path inflation, In Proceedings of ACM SIGCOMM Conference, 2003.
[10] C. Labovitz, A. Ahuja, A. Bose, and F. Jahanian, Delayed Internet routing convergence, In Proceedings of ACM SIGCOMM Conference, 2000.
[11] One way transmission time, ITU-T Recommendation G.114, 2000.
[12] L. Tang, Y. Chen, F. Li, H. Zhang, and J. Li. Empirical study on the evolution of PlanetLab. In Proceedings of 6th International Conference on Networking, 2007.
[13] L. Tang, Z. Sun, Z. Chen, and J. Li. On the Feasibility of Enhancing Interactivity for Real-time Communications using Overlay Routing. In Proceedings of 3rd International Conference on Networking and Services, 2007.
[14] S. Tao, K. Xu, A. Estepa, T. Fei, L. Gao, R. Gu´erin, J. Kurose, D. Towsley, and Z.-L. Zhang, Improving VoIP quality through path switching," In Proceedings of IEEE Infocom Conference, 2005.
[15] PlanetLab. http://www.planet-lab.org/.
[16] p2psim. http://pdos.csail.mit.edu/p2psim/.
[17] All-Pairs-Pings: http://pdos.csail.mit.edu/~strib/pl_app/.