A Measurement Study of Internet Delay Asymmetry

Abhinav Pathak[1], Himabindu Pucha[1], Ying Zhang[2], Y. Charlie Hu[1], Z. Morley Mao[2]

[1] Purdue University
[2] University of Michigan
pathaka@purdue.edu, hpucha@purdue.edu, wingying@umich.edu,
ychu@purdue.edu, zmao@umich.edu

Abstract. RTT has been widely used as a metric for peer/server selection. However, many applications involving closest peer/server selection like streaming, tree-based multicast services and other UDP and TCP based services would benefit more from knowing one-way delay (OWD) rather than RTT. In fact, RTT is a compromise solution in most cases and is used to infer forward and reverse delays by many protocols and applications. They assume forward and reverse delay to be equal to half of RTT.

In this paper we compare and contrast one way delays and corresponding RTTs using a wide selection of routes in the Internet. We first measure the extent and severeness of asymmetry in forward and reverse OWD in Internet. We then attempt to isolate the causes of OWD asymmetry by correlating OWD asymmetry with the route asymmetry. Finally, we investigate the dynamics of delay asymmetry: There exists a weak correlation between the fluctuation of RTT and OWD and a strong correlation between OWD change and the corresponding route change.

1 Introduction

Today's Internet is rife with several wide-area network applications: real-time applications such as voice over IP [1] and multicast streaming applications [2–5], data transfer applications that perform locality-aware redirection and server selection [6], and services such as proximity-aware DHTs [7, 8] and positioning systems [9–11]. A common thread in all these applications is the requirement to perform proximity measurements. For example, in multicast applications, proximity is used to choose a suitable parent/child in the tree. In positioning systems, proximity to landmarks is used for localization.

For some applications, the proximity of interests can be measured using the round-trip time (RTT) between two end hosts (say A and B), defined as the sum of forward delay from A to B and the reverse delay from B to A. For example, if the interaction between remote hosts typically involves only one or a few request and reply messages, for example, a DNS lookup, or a small HTTP document download, then RTT is good indication of the completion time of the interaction. For other applications, however, the proximity of direct relevance is the one-way delay (OWD) from the client to the servers/peers or along the other direction, rather than RTT. An asymmetry in OWDs could hurt such applications. For example, in multicast streaming applications, since the data always flows from a parent node to a child node in the overlay multicast tree,

optimizing the OWD from the parent to the child is more beneficial. As another example, ACK/NAK data systems such as the Transmission Control Protocol (TCP) estimate the available bandwidth of the unidirectional route from the sender to the receiver using the round-trip time as an approximation. If the reverse path taken by ACK packets has a much larger delay than the forward path delay, TCP can end up using more network resources than it should [12]. Other applications that depends on OWDs include online multiplayer games where an asymmetry in delay could create a bias, video conferencing applications, Internet distance prediction, etc.

In practice, measuring the OWD between two end hosts, however, faces two major obstacles. First, it requires strict time synchronization between the two hosts. Second, it requires access to both end hosts as there is no standard daemon in operating systems that measures and reports OWD. As a result, the standard practice in almost all applications that rely on proximity information is to measure RTT and operate under the assumption that OWD is half of RTT.

In this work, we investigate the validity and implications of this premise (and common practice) via a comprehensive measurement study: We first measure the extent and severeness of asymmetry in forward and reverse OWD between several pairs of Internet hosts. We then attempt to trace the reasons for the observed OWD-RTT relationship. Further, we investigate the dynamics of the OWD-RTT relationship as both OWD and RTT change. Our main findings are as follows: (1) Asymmetry between the forward and reverse delays is quite prevalent. (2) Asymmetry in delay can be attributed at least in part to the asymmetry in routing paths. (3) Delay asymmetry is dynamic - with progression of time, delay asymmetry varies. To track how the asymmetry varies as RTT and route changes happen, we make the following observations: During an intra-AS path change in most cases, the forward and reverse delay changes equally. During an inter-AS path change, either both forward and reverse delay change equally (keeping asymmetry constant), or only forward delay change contributes to RTT change (changing asymmetry).

2 Methodology

2.1 Tools

Owping: We use owping [13] for measuring OWDs between a source-destination pair. Owping is an implementation of One-way Active Measurement Protocol (OWAMP) [14]. It requires access to both end hosts between whom the delay is to be measured. The destination node runs one-way ping server, "owampd" (owamp daemon), which listens for client requests for conducting one-way ping measurements. The source node initiates the measurement using owping (client). For every measurement, the source and destination exchange 10 probe packets.

Owping requires time synchronization between the end hosts running the measurements. OWDs between Internet hosts range from a few milliseconds to hundreds of milliseconds. Even a minimal clock drift could result in inaccuracy in OWD measurements. To capture clock drifts (difference between node's clock and NTP's clock), before reporting any "time" measurements, the owping tool gets the current clock drift of the local node from the NTP daemon running on the system. The clock drift is adjusted in the time reported by the system. NTP daemon also reports error estimate (confidence value) of the clock. Error estimate gives the confidence range (in milliseconds) of time

reported by NTP. The source marks its current time (adjusted with clock drift) on the packet along with its current error estimate with respect to NTP before sending the packet. Upon receiving the packet, the destination subtracts the time on the packet from its current time (fetched using kernel timestamp of packet and then corrected with its clock drift). It also notes down error estimate at its end due to NTP. The overall error estimate for a single probe packet is calculated by adding up the corresponding error estimates of both ends. For all the 10 packets, OWD and error estimates are noted down. The minimum, mean and maximum values of the 10 OWD are reported, along with the maximum error estimate of the 10 error estimates.

Paris-traceroute: To measure the route between a pair of nodes, we used Paris traceroute [15] which allows more accurate measurement of a route than the original traceroute by avoiding a number of routing anomalies.

2.2 Testbed and Trace Collection

Since owping requires access to both end hosts for which the delay is measured, we use the Planetlab [16] testbed for our measurement study. Planetlab contains nodes belonging to both research/education networks (GREN) and commercial networks. A recent study [17] has shown that the network properties of the paths between two hosts in GREN (denoted as G2G - Gren-to-Gren) can be very different from those in the commercial networks. The same study also showed that if at least one end of the path is in the commercial network (G2C or C2G), then the network properties remain similar compared to when both ends are in the commercial network (C2C). Hence, we perform this study by separating the network paths into all GREN and commercial paths (G2C+C2G+C2C). We chose 180 GREN nodes and 25 commercial nodes.

We collected traces from 12th April 2007, for a period of 10 days. Traces consist of back-to-back traceroutes and OWD measurements. One measurement round consists of a traceroute and a OWD measurement to every other node in our set and was repeated after every 20 minutes. The data collected was stored on a central node for parsing and further processing. We continuously monitored over 10000 Internet paths during this period. We conducted over 5 million traceroutes and an equal number of OWD estimations during the specified period. We minimized the error induced due to loaded planetlab nodes [18] by taking minimum of 10 readings reported by owping and using kernel timestamps.
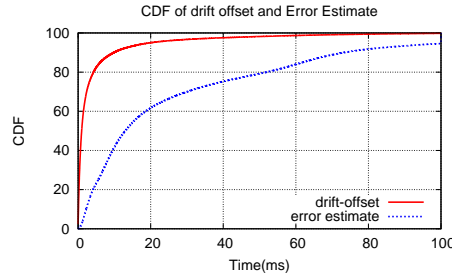

Fig. 1. CDF of drift offset and error estimate among 205 nodes in Planetlab

2.3 Trace Pruning

Accurate one-way ping measurements critically depends on end nodes synchronization. When a host reboots, its clock is not synchronized and hence its measurements are not usable. However, after booting up, it contacts an NTP server and is configured to correct its clock [19]. This adjustment consumes some time after which the node is synchronized with NTP. Typically, the uptime of Planetlab nodes is large. However, when a node goes down, we remove it from our measurements. We also eliminate nodes with large error estimates. Figure 1 plots the CDF of clock drift and the maximum error estimate for Planetlab nodes. We observe that Planetlab nodes do not drift much from their NTP servers. 60% of the nodes drift less than 2 milliseconds. We also observe that the maximum error estimated by NTP is lower than 10 ms for about 40% of the nodes. As a first cut we chose 10ms error estimate threshold and pruned our node selection by eliminating nodes that deviated more. After this filtering, we had 82 GREN nodes and 12 commercial nodes. When computing any ratios using absolute forward and reverse delays we limit the values of error estimate. For example, in section 3 wherever we took ratios involving OWD, such as forward delay by RTT, we made sure that the error estimated in forward delay stays below 3% of the measured RTT. In section 4 we deal with change in forward and reverse delays at a node. We correlate them to change of RTT. Whenever we take a difference, of say forward delay, the error in the value gets nullified. This way we do not need additional data pruning.

2.4 Metrics

We ultimately relate delay asymmetry on the Internet to other asymmetric properties like network path. In this section we define metrics that we used to quantify the level of path asymmetry among the measured routes. Our metric needs to adequately capture the properties of an Internet path responsible for inducing delays in the network. The packet delay introduced by the path depends on the intermediate routers and ASes through which the packet travels. We compare two metrics to characterize these properties - AS level path asymmetry and router level path asymmetry. **AS level path asymmetry** captures the dissimilarity among ASes in the forward and the reverse paths. We quantify AS level path asymmetry using AS level path similarity coefficient (γ) as follows: For every source destination pair, let the set of ASes in the forward path (P_f) and the reverse path (P_r) be A and B respectively. The similarity coefficient γ_{P_f, P_r} is calculated as

$$\gamma_{P_f, P_r} = \frac{|A \cap B|}{|A \cup B|} \tag{1}$$

Router level path asymmetry captures the dissimilarity between the forward and the reverse path at the IP-level hops. To determine the router level asymmetry in the forward and reverse path, we can not simply use IP addresses in the traceroutes in forward and reverse directions because many routers have different interfaces to handle traffic in different directions. Most of these interfaces lie in the same /24 prefix. Grouping IP addresses of intermediate routers obtained with same /24 prefixes into one loses the hop count information of the path. To avoid these problems we use the following approach: In each path we took the /24 prefixes of the IP addresses of the intermediate routers. If there exists more than one interface belonging to the same /24 prefix in the path, we counted them differently. Specifically we concatenated an increasing counter per /24
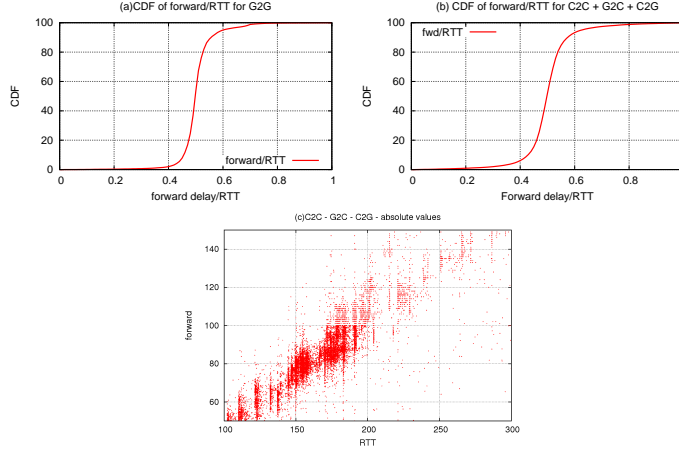
Fig. 2. CDF of forward OWD as a fraction of RTT for different RTTs occurrences.

prefix to every duplicate of that prefix encountered. This was done for both forward and reverse paths. We construct set A using this information in forward path and B from reverse path. With these definitions of A and B we compute router level similarity coefficient using equation (1).
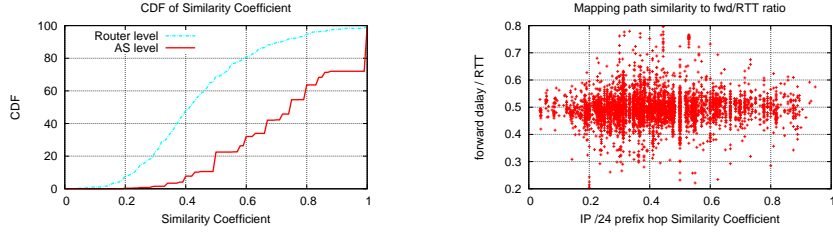
3 Delay Asymmetry

3.1 One-Way vs. Round-Trip

OWD in forward and reverse directions add up to give RTT, with a general perception that the forward and reverse delays are equal. This section investigates the extent and the severeness of delay asymmetry among GREN and commercial paths individually. To that end, we observe the correlation between OWD and RTT as shown in Figure 2.

Figure 2(a) shows that paths in G2G are largely symmetric with respect to their delays (CDF remaining close to 0.5 shows symmetry). On the other hand, Figure 2(b) indicates that asymmetry is prevalent in commercial networks. In fact, the magnitude of asymmetry (the ratio of OWD in forward direction to RTT) varies from values below 0.4 to those above 0.6. Figure 2(c) augments 2(b) and shows the absolute values of forward delay and RTT for non-G2G paths. Figure 2(c) shows the magnitude of asymmetry. For example, we take the point where RTT is 150ms and forward delay is 60ms. This case would correspond to 40% of forward delay ratio. The reverse path delay accounts for 90ms. This results in 30ms of delay asymmetry in the forward and reverse paths. We see that asymmetry in delay indeed exists in today's Internet especially in commercial networks.

3.2 Asymmetry in One Way Delay

We now dig a little deeper to understand the origin of this asymmetry in delay values. Intuitively, if the forward and the reverse paths between a source-destination pair are different, we can expect the corresponding properties to vary too. Path asymmetry is a well-known fact prevalent in the Internet. Many previous studies have found that a large amount of asymmetry exists in Internet routes [20–23]. In this section we study whether there exists any correlation between route asymmetry and delay asymmetry.

(a) CDF of AS level and router level similarity coefficient in forward and reverse path of all routes observed.

(b) Scatter plot of router level similarity coefficient for each route vs. ratio of forward delay and RTT.

Fig. 3. Comparing path asymmetry with delay asymmetry

Figure 3(a) plots the CDF of AS level and router level similarity coefficients (defined in section 2.4) for all the possible routes that were found in our trace. Figure 3(a) shows that 32% of paths have AS similarity coefficient of less than 0.6, while 81% of routes have router-level asymmetry coefficient of less than 0.6. The graph shows, as expected, that router level asymmetry is more prevalent than AS level asymmetry. Hence we use router level similarity coefficient to characterize path asymmetry.

To find correlation between delay and path asymmetry we plot delay fraction, the ratio of forward delay by RTT, as a function of router level similarity coefficient for each route observed. Figure 3(b) shows the correlation. From the figure we see that the delay fraction remains close to 0.5 when the router level similarity coefficient is close to unity. In such cases, the forward and reverse paths traverse nearly the same set of routers and experience equal delays. They contribute equally to RTT. In cases when router level similarity coefficient is not close to unity the delay fraction fluctuates from 0.3 to 0.7. This gives us an indication that if there exists a significant router level asymmetry, the forward and reverse OWD could be significantly different. In summary router level asymmetry does not imply delay asymmetry where as delay asymmetry implies router level asymmetry.
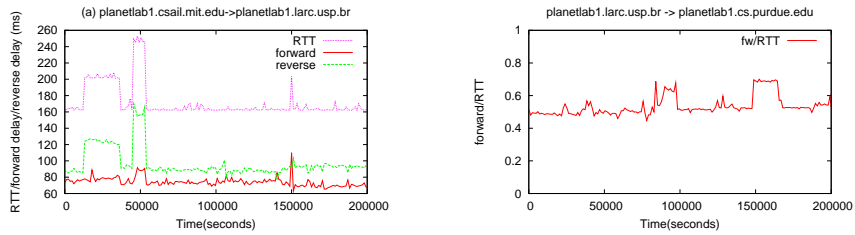
(a) Absolute forward OWD and RTT for selected source-destination pair.

(b) forward/RTT over time for a particular source destination pair

Fig. 4. Dynamics of delay asymmetry

4   Dynamics of Delay Asymmetry

Although existence of delay asymmetry is of interest, another intriguing question is: Is the delay asymmetry for a given source-destination pair constant across time? If delay

Legend	RTT Changes	Forward Changes	Reverse Changes	Explanation
A	Yes	No	Yes	RTT changes due to a change in reverse delay but forward delay is unchanged
B	Yes	Yes	No	RTT changes due to change in forward delay but reverse delay is unchanged
C	Yes	Yes	Yes	RTT changes along with changes in forward and reverse delays
D	No	Yes	Yes	RTT does not change but both forward and reverse delay changes (equally and opposite in sign)

Table 1. Various possibilities observed that change delay asymmetry

asymmetry remains constant then we could do a one-time measurement and tune applications accordingly. If not, how does it vary? To get an immediate insight into this question, we randomly picked a source-destination (Brazil-Purdue) pair and observed how the delay fraction varies over time (Figure 4(b)). From the figure we see that delay fraction changes noticeably at two instances of time (just before 100000 seconds and just after 150000 seconds). The delay fraction at these instances rises to show the increase in contribution of forward delay to RTT. Having found evidence of dynamics in delay asymmetry, we now attempt to understand this effect in detail. The delay fraction (forward delay/RTT) changes when either the forward delay, RTT or both change.

Table 1 categorizes the various possibilities that can change delay asymmetry. From the table we see four reasons can cause delay asymmetry change. As an example, consider Figure 4(a) which plots RTT, forward and reverse delays between a node at MIT and a node at Brazil. We see two step ups in RTT value at about 25000 second and 50000 seconds and both are contributed by the increase in reverse path delay. The delay in the forward path remains relatively stable. For this scenario the delay fraction, the ratio of forward by RTT, starts at about 0.5, drops to 0.36 around the first step up and then further drops down to about 0.34 around the second step up. This scenario depicts legend A of Table 1.

To get a better picture of the prevalence of the above scenarios, we log the fluctuations in RTT values across our trace: Whenever RTT fluctuated by 2% for any source/destination pair, we note down the corresponding fluctuation in forward and reverse delays. Figure 5(a) plots a scatter plot to show the correlation between forward OWD change and RTT fluctuation. We see that the plot can be broken down into 4 major regions. The first region is parallel and close to x-axis (line y=0) (legend A of table 1), the second region along y=x line (legend B of table 1), a third region along y=0.5x line (legend C of table 1), and the fourth region parallel and close to y axis (x=0 line) (legend D of Table 1).

Thus, there is conclusive evidence that delay asymmetry is a dynamic property. Delay asymmetry changes when delays change. [24] points out two reasons for delay changes - path change and transient congestion. Out of these two major reasons of delay changes, we focus on delay changes caused due to a path change in forward direction. These path changes are measured using periodic traceroutes from source to destination. Traceroute from a source to destination depicts the path in the forward direction. Using repeated traceroutes we find out occurrence of a route change in forward direction (at a granularity of 20 minutes). At the same time we traceroute the path in the reverse direc-
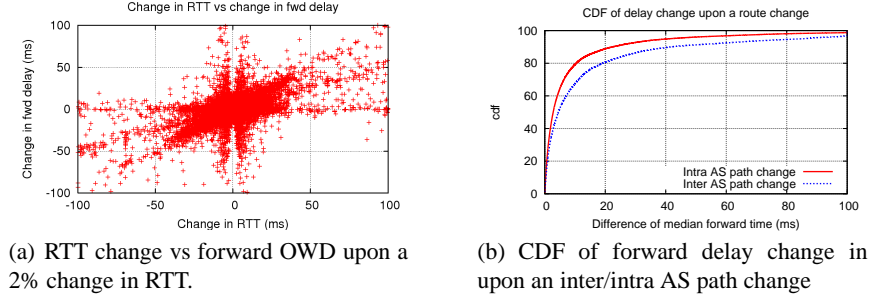
(a) RTT change vs forward OWD upon a 2% change in RTT.

(b) CDF of forward delay change in upon an inter/intra AS path change

Fig. 5. Dynamics of delay asymmetry

tion and also measure one way delays. We divide the path change in forward direction into two categories, inter AS path change and intra AS path change.

Figure 5(b) shows by how much forward delay changes upon a change in forward path. For every path change we observed in our measurement, we classified the path change into inter AS path change if the new path has a different AS level path then the previous path or intra AS path change, if there is a path change internal to an AS. We see that 80% of intra AS path change results in change in forward delay by less than 10 milliseconds. Upon an inter AS path change, about 80% of times the forward delay changes by less than 20 milliseconds.

A path change might change the delay asymmetry for a route. To find out impact of path changes on change in OWD and RTT, for every inter and intra AS path change that we observed, we plot a scatter plot between the absolute change in forward delay and absolute change in reverse delay. Figure 6(a) shows correlation between forward and reverse delay changes upon a forward intra AS path change. We see that there is a strong correlation between the attributed delay change values. In most of the cases, the change in forward and reverse delay equally contribute to the change in RTT (the patch of dots along the y=x line). This implies that when we notice a path change in forward direction, in most of the cases reverse direction path also changes.

Figure 6(b) shows correlation between forward and reverse delays when there is an inter AS forward path change. There can be two cases here. The first, if AS change takes place only in the forward direction and the second, if the AS change in forward path also affects reverse path. Inter AS path changes are less frequent than intra AS path changes. The scatter plot in figure 6(b) shows two faint regions, one along the line y=x and the second region parallel and close to x-axis. The first region signifies of AS path change when forward and reverse delays equally contribute to RTT. This region captures the situation when forward AS path change also affects path in the reverse direction. The second region depicts the situation when AS level path changes in the forward direction, forward delay is changed but reverse delay is not affected. In this case, the change in RTT is contributed solely by forward delay.

In summary, we have identified causes for delay symmetry changes based on routing changes. The property of delay asymmetry is found to be a dynamic property that varies depending on routing dynamics. As expected, inter AS path changes contribute to larger delay changes compared to intra AS path changes. Moreover, intra AS path changes tend to have similar effect in terms of delay changes for both forward and reverse paths.
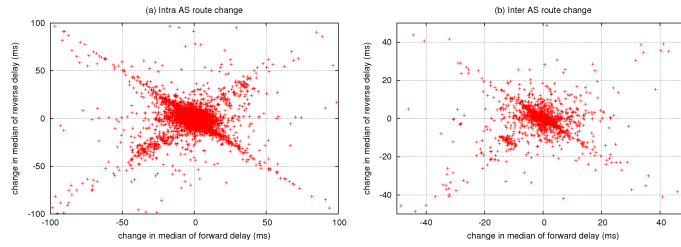
Fig. 6. Scatter plot: Reverse vs forward delay change for Intra/Inter AS path changes.

5 Related Work

Paxson [20] was among the first to study asymmetry in Internet routes. He et. al. [23] quantified the level of path asymmetry that exists in the current Internet. They conducted a systematic study of quantifying asymmetry in academic and commercial nodes. Our work is the first to quantify the delay differences as a result of path asymmetry. Our measurement follows the RFC 2679 [25] which defines a metric for OWD of packets across Internet paths. There have been proposals for measuring OWD using mathematical heuristics based on relationships between a sequence of back-to-back one-way packet delays. Choi and Yoo [26] proposed a scheme to derive OWD. However, they did not quantify the delay asymmetry or its dynamics in the current Internet. Several work [12, 27] have studied performance degradation of TCP due as a result of asymmetric network conditions.

6 Conclusion

In this paper, we studied delay asymmetry that exists in the current Internet. We found that commercial networks exhibit higher levels of asymmetry than education and research networks. We found a weak correlation between router-level path asymmetry and delay asymmetry. We then studied how delay asymmetry changes over time as path change occurs. We found that any of the four scenarios could take place (a) forward delay change equals RTT change with no change in reverse delay, (b) forward and reverse delays change equally contribute to a change in RTT, (c) reverse delay change contributes to entire RTT change with negligible forward delay change, and (d) forward delay and reverse delay change equally but in opposite directions, resulting in no effect on RTT. We found out properties of delay changes upon an inter- and intra-AS route change. Our work provides important foundations to enable applications to predict OWD changes as a result of possible routing changes and more accurately infer OWD values from RTT measurements.

Our findings suggest that proximity-based applications that need information about OWD can benefit from accurate OWD measurement as opposed to using half of RTT as an approximation. However, OWD measurement requires the cooperation of both end hosts. One possible solution to this is to incorporate OWD measurement software as a daemon in commodity OSes similarly as the ICMP echo daemon for normal ping.

References

1. : Skype http://www.skype.com/.

2. hua Chu, Y., Rao, S.G., Zhang, H.: A case for end system multicast. In: Proc. of ACM SIGMETRICS. (2000)

3. Zhang, B., Jamin, S., Zhang, L.: Host Multicast: A Framework for Delivering Multicast To End Users. In: Proc. of IEEE INFOCOM. (June 2002)

4. Banerjee, S., Bhattacharjee, B., Kommareddy, C.: Scalable application layer multicast. In: Proc. of ACM SIGCOMM. (2002)

5. Castro, M., Druschel, P., Kermarrec, A.M., Nandi, A., Rowstron, A., Singh, A.: SplitStream: High-Bandwidth Multicast in Cooperative Environments. In: Proc. of ACM SOSP. (2003)

6. : Akamai http://www.akamai.com/.

7. Rowstron, A., Druschel, P.: Pastry: Scalable, Distributed Object Location and Routing for Large-Scale Peer-to-peer Systems. In: Proc. of Middleware. (2001)

8. Zhao, B.Y., Huang, L., Stribling, J., Rhea, S.C., Joseph, A.D., Kubiatowicz, J.: Tapestry: A Resilient Global-Scale Overlay for Service Deployment. IEEE JSAC (2004)

9. Ng, T.S.E., Zhang, H.: Predicting Internet Network Distance with Coordinates-Based Approaches. In: Proceedings of IEEE INFOCOM. (June 2002)

10. Dabek, F., Cox, R., Kaashoek, F., Morris, R.: Vivaldi: A Decentralized Network Coordinate System. In: Proceedings of ACM SIGCOMM. (August 2004)

11. Francis, P., Jamin, S., Paxson, V., Zhang, L., Gryniewicz, D., Jin, Y.: An Architecture for a Global Internet Host Distance Estimation Service. In: Proceedings of IEEE INFOCOM. (March 1999)

12. Balakrishnan, H., Padmanabhan, V.N., Katz, R.H.: The effects of asymmetry on tcp performance. In: Proc. of ACM MobiCom. (September 1997)

13. : One way ping http://e2epi.internet2.edu/owamp/.

14. Shalunov, S., Teitelbaum, B., Karp, A., Boote, J., Zekauskas, M.: A One-way Active Measurement Protocol (OWAMP). RFC 4656 (Proposed Standard) (September 2006)

15. Augustin, B.: et al., Avoiding traceroute anomalies with Paris traceroute. In: Proc. of IMC. (2006)

16. : Planet lab http://www.planet-lab.org/.

17. Pucha, H., Hu, Y.C., Mao, Z.M.: On the Representativeness of Wide Area Internet Testbed Experiments. In: Proc. of ACM IMC. (2006)

18. Sommers, J., Barford, P.: An Active Measurement System for Shared Environments. In: Proceedings of IMC. (October 2007)

19. Mills, D.L.: RFC 1305: Network time protocol (version 3) specification, implementation (March 1992) Obsoletes RFC0958, RFC1059, RFC1119 Status: DRAFT STANDARD.

20. Paxson, V.: End-to-end routing behavior in the Internet. In: Proc. of ACM SIGCOMM. (1996)

21. Allman, M., Paxson, V.: On estimating end-to-end network path properties. In: Proc. of SIGCOMM. (1999)

22. Savage, S., Collins, A., Hoffman, E., Snell, J., Anderson, T.E.: The end-to-end effects of Internet path selection. In: Proc. of SIGCOMM. (1999)

23. He, Y., Faloutsos, M., Krishnamurthy, S., , Huffaker, B.: On routing asymmetry in the internet. In: Proceedings of IEEE Globecom, 2005. (2005)

24. Pucha, H., Zhang, Y., Mao, Z.M., Hu, Y.C.: Understanding Network Delay Changes Caused by Routing Events. In: Proc. of ACM SIGMETRICS. (2007)

25. Almes, G., Kalidindi, S., Zekauskas, M.: A One-way Delay Metric for IPPM. RFC 2679 (September 1999)

26. Choi, J.H., Yoo, C.: One-way delay estimation and its application. Computer Communications **28**(7) (2005) 819–828

27. Balakrishnan, H., Padmanabhan, V., Fairhurst, G., Sooriyabandara, M.: TCP performance implications of network path asymmetry. In: IETF RFC 3449. (2002)