

# A Study of Internet Packet Reordering<sup>\*</sup>

Yi Wang<sup>1</sup>, Guohan Lu<sup>2</sup>, and Xing Li<sup>1</sup>

<sup>1</sup> Department of Electronic Engineering, Tsinghua University  
Beijing, 100084, P. R. China  
wangyi@ns.6test.edu.cn  
xing@cernet.edu.cn

<sup>2</sup> China Education and Research Network (CERNET)  
Beijing, 100084, P. R. China  
lvguohan@tsinghua.org.cn

**Abstract.** Packet reordering is a well-known phenomenon that the order of packets is inverted in the Internet. Previous studies indicates reordering can affect the performance of both the network and the packets receiver. Nevertheless, they get different results about the prevalence of reordering in the Internet. In this paper, we firstly present a methodology for single-point reordering measurement at a TCP receiver, including the algorithm and its implementation. Then we show the results of our three-week observation of reordering from a set of 10,647 Internet Web sites in China. In addition, we discuss a method to distinguish reordering and loss by making use of the distribution of their time lag and packet lag. Finally, we study the pertinence of sites experiencing reordering according to the network topology and propose a novel and relatively reliable approach to infer reorder-generating spots in the Internet.

## 1 Introduction

Packet reordering in the Internet is a well-known phenomenon. It is deceptively simple that packets can be reordered due to multi-path routing or parallelism at the routers. However, packet reordering is practically challenging to study, since it is a silent problem leaving little to no trace.

The IP layer in the Internet just provides a “best effort” datagram service. Although TCP is a reliable higher-layer protocol, packet reordering can affect it’s performance and the efficiency of packet receiver for several reasons:

- **Causes Unnecessary Retransmission:** When the TCP receiver gets packets out of order, it sends duplicate ACKs to trigger fast retransmit algorithm at the sender [12]. These ACKs (3 or more) makes the TCP sender infer a packet has been lost and retransmit it. If the temporary sequence number gap is caused by reordering, then the duplicate ACKs and the fast retransmission are unnecessary and a waste of bandwidth.

---

<sup>\*</sup> This work is supported by Cisco University Research Program.

- **Limits Transmission Speed:** When fast retransmission is triggered by duplicate ACKs, the TCP sender assumes it is an indication of network congestion. It reduces its congestion window (*cwnd*) to limit the transmission speed, which needs to grow larger from a “slow start” again. If reordering happens frequently, the congestion window is at a small size and can hardly grow larger. It results in a limited speed of packets transmission, and hence a throughput degradation [14].
- **Reduce Receiver’s Efficiency:** Since the TCP receiver has to hand in data to the upper layer in order, when reordering happens, the receiver has to buffer all the out-of-order packets until getting all packets in order. Meanwhile, the upper layer gets data in burst rather than smoothly, which also reduces the system efficiency as a whole.

As the load of Internet grows, packet transmission equipments that do not guarantee FIFO are more and more used. It is worthwhile to know the frequency and magnitude of packet reordering in the Internet. Previous studies get discrepant results of the prevalence of reordering [1, 2, 3]. The reason partially lies in the methodological differences, and partially lies in the fast growth of the Internet. What is more, previous work on reordering mainly focused on the causes, dynamic characters and improving TCP performance in the face of reordering beyond measurement. No work has been done about correlation between reordering and the network topology to our knowledge. In this paper, we design a methodology which is not only suitable to common measurement of reordering, but also convenient for studying the above correlation.

The remainder of the paper is organized as follows. In Sect. 2, we review the related work. We propose our measurement methodology in Sect. 3, followed by Sect. 4 that shows our measurement results. Our novel approach to infer reordering-generating spots in the Internet is presented in Sect. 5. Finally, Sect. 6 concludes the whole paper.

## 2 Related Work

Previous studies of packet reordering can be approximately divided into two categories: general measurement study, which includes measurement methodology and experiment in the Internet; and specific topics on reordering, such as the causes, measurement techniques and metrics, improvement of TCP performance in the face of reordering, etc.

The first category of study is the fundamental of understanding packet reordering. Paxson’s 1997 study [1] is based on a series of measurements taken between 35 Internet sites by transferring 100Kbyte TCP bulks on 1994 and 1995 separately. Paxson reports during two measurement periods, 36% and 12% of sessions experienced at least one reordering event respectively, and 2.0% and 0.26% of packets were reordered. Bennett et al’s study work [3] in the year 1997-1998 at MAE-East network use a different approach in which they measure reordering by sending back-to-back ICMP-ping packets and evaluate the response. They

report that over 90% of packets were reordered during their two measurements of 140 Internet hosts. While Jaiswal et al's 2002 measurement in Sprint IP backbone observe a relatively lower rate of reordered packets of approximately 5% [2]. Instead of measuring end-to-end probe traces at the sender or receiver, they measure reordering at a single point within the backbone.

In the second category of study, Bennett et al's 1999 paper [3] attributes most reordering to "local parallelism". Liu's 2002 paper [5] does further discussion about the packet level parallelism. Bellardo and Savage describe a set of measurement techniques that can estimate one-way end-to-end reordering rates [4]. There are also some studies on modifications to TCP aiming better tolerance of reordering [8, 9, 10, 11].

### 3 Methodology

As mentioned in Sect. 1, what we are interested in is not only the frequency and magnitude of packet reordering in the Internet, but also the relationship between reordering and network topology. We propose a novel single-point and easy-implemented measurement methodology that can meet both the two aims without requiring either the control of both ends of the connections (e.g., see [1]) or the privilege of accessing the backbone (e.g., see [2]).

#### 3.1 Measurement Environment

Our measurement uses a host in the CERNET<sup>1</sup> as the measurement point. We choose 10,647 web sites in China<sup>2</sup> as our data source, since the WWW (to be more precise, the HTTP on port 80) is the most widely used service in the Internet according to [13]. To the 10,647 web sites, we firstly did web page crawling (using *wget*) and measured forward-path<sup>3</sup> reordering twice a day from May 3 - May 12, 2003. Then we divided these sites into two categories: *Reorder Sites* (591 sites that experienced reordering at least once) and *Ordinary Sites* (10,056 sites that experienced no reordering). From May 16 - June 5, 2003, we did consecutive measurement comparison between the two categories. Every 3 hours, we measured reordering by crawling web pages from all the Reorder Sites. At the same time, we randomly crawled Ordinary Sites of the same number. When reordering was observed from an Ordinary Site, it was moved into the Reorder Sites category before the next measurement began.

#### 3.2 Measurement Environment

Generally, we say that a packet is out-of-sequence if its Seq (TCP sequence number) is less than that of a previous received packet in the same connection. An

<sup>1</sup> China Education and Research Network

<sup>2</sup> These web sites are routed inside China according to the IP blocks routed inside China on March 2003 announced by CERNIC (<http://www.nic.edu.cn>).

<sup>3</sup> Define it as the direction from the web sites to the measurement host. The opposite direction is called the backward-path direction.

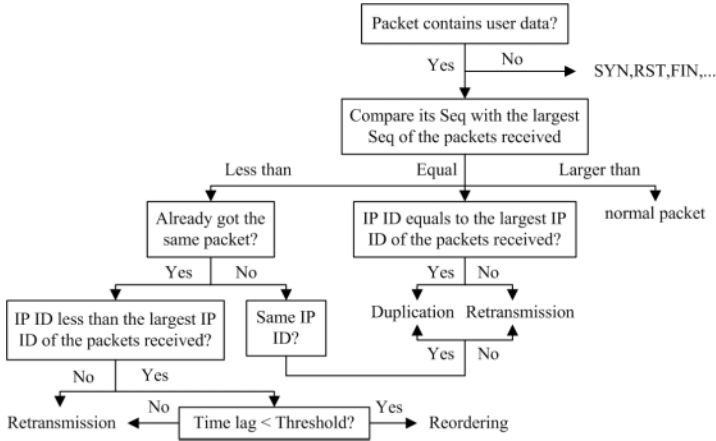


Fig. 1. Process of the reorder-judging algorithm

out-of-sequence packet could be the result of retransmission, network duplication or network reordering. These three causes are essentially different (see [2]). In this paper, not like many of previous studies, we focus on the network reordering which is mainly caused by parallelism within a router or a route change. Figure 1 shows the reorder-judging algorithm we proposed at the TCP receiver, which can distinguish most of out-of-sequence packets for different causes. It is based on Seq (TCP sequence number), IP ID and the time lag between packets [6]. Since the IPID of TCP wraps to 0 when monotonically increases to 65535, it is possible that the IPID of a retransmitted packet from a busy site is less than the previous lost one's. However, the time lag of the retransmitted packet must be much larger than a reordered packet because of the fast retransmission algorithm. So we set a threshold of 300ms to the time lag to distinguish reordered packet and retransmitted packet with IPID wrapped<sup>4</sup>.

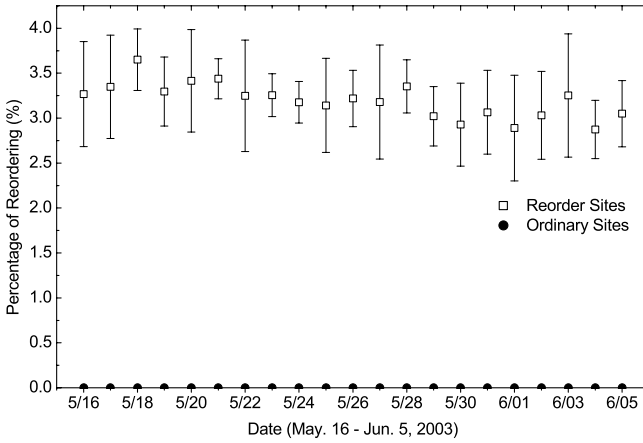
## 4 Results

In this section, we show the results of our measurement and discuss the approach to distinguish reordering and loss.

### 4.1 Measurement Data

In the three-week period (May 16 – June 5, 2003), we traced 208 thousand connections with totally 3.3 million data packets. 3.197% of all the packets were reordered. 5.79% of all 10,647 web sites (that is, 616 Reorder Sites) experienced

<sup>4</sup> Our measurement data shows it is a rare instance and the time lags of this kind of retransmitted packet are usually larger than 500ms. Figure 4 in Sect. 4.2 also shows 300ms is long enough for almost all the reordered packets to arrive.



**Fig. 2.** Distribution of packet reordering from May 16 – June 5, 2003

**Table 1.** Reordering frequency of Reorder Sites

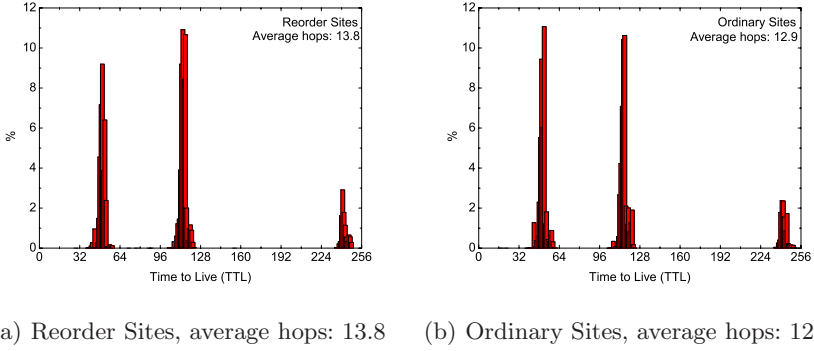
Reordering Freq.	>90%	80%–90%	70%–80%	60%–70%	50%–60%
Number of Sites	66	50	31	22	38
Percentage(%)	10.71	8.12	5.03	3.57	6.17
Reordering Freq.	40%–50%	30%–40%	20%–30%	10%–20%	0%–10%
Number of Sites	39	55	65	72	178
Percentage(%)	6.33	8.93	10.55	11.69	28.90

reordering at least once. Figure 2 shows the distribution of packet reordering over the entire duration of our measurement.

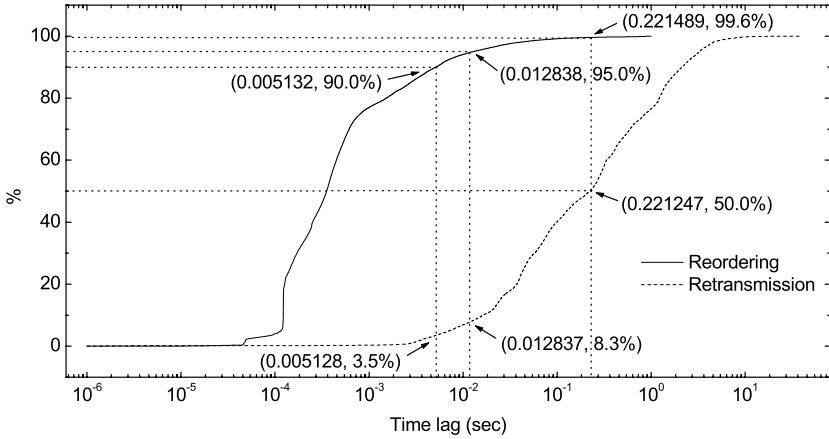
The discrepancy of reordering rate between the two site categories is huge and relatively steady. Reordering rate of Reorder Sites is between 2.39%–4.27% with a mean of 3.197%, while the reordering rate of Ordinary Sites is always below 0.14% with a mean of 0.017%. This discrepancy indicates that reordering is strongly site-dependent and occurs mainly in some certain parts of the Internet.

Table 1 summarizes the reordering frequency of the 616 Reorder Sites. About 20% of the Reorder Sites are with a reordering frequency higher than 80%. These sites contribute the bulk of reordering in our experiment.

Figure 3 shows the distribution of TTL (time-to-live) values of Reorder Sites and Ordinary Sites. Since TTL values are usually set to a few well-known values such as 64, 128 and 256, we can easily infer the distance from a web site to the measurement host in terms of the number of router hops. We find Reorder Sites (with average hops value 13.8) tend to be farther away from the measurement host than the Ordinary Sites (with average hops value 12.9).



**Fig. 3.** Distribution of TTL

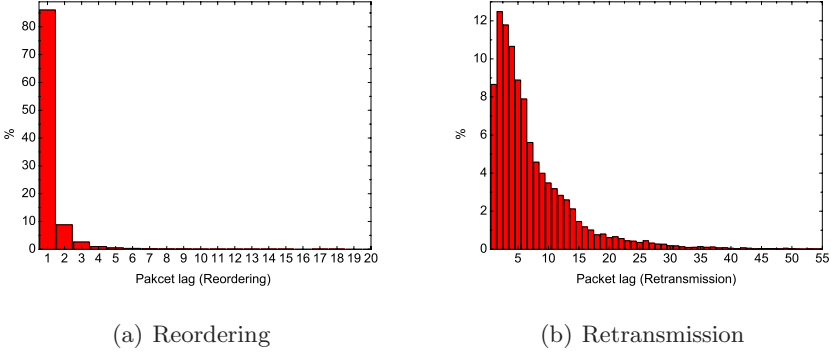


**Fig. 4.** Cumulative distribution of time lag of reordering and retransmission

### 4.2 Distinguishing Reordering and Loss

One of the main reasons that packet reordering affects the TCP performance is that, TCP would mistake reordering for loss when meeting sequence hole at the receiver. Since loss can not be confirmed until retransmitted packet arrived, we studied the time lag and packet lag of both packet reordering and retransmission. What we found indicates that we can distinguish them by setting certain threshold.

Figure 4 shows the cumulative distribution of time lag of both reordering and retransmission. 90% of reordered packets arrive at the receiver with time lag less than 5.1 ms, while only 3.5% of retransmitted packets arrive then. We find 12.8 ms is a relative good threshold in our experiment: 95% of reordered packets arrived but only 8.3% of retransmitted packets arrive.



**Fig. 5.** Distribution of packet lag

Figure 5 shows the packet lag of reordering and retransmission. 86.5% of reordered packets left behind only 1 packet and 95.3% of reordered packets left behind within 2 packets. Packet lag of retransmission has a dispersed distribution and a larger mean. About 78.8% of retransmitted packets left behind with a lag of 3 or more packets.

From the discussion about packet lag, we can evaluate the impact of reordering on TCP performance. Since over 95% of reordered packets are with a packet lag less than 3, there is only a little probability that reordering would trigger the TCP fast retransmit algorithm, which is consistent with [7]. However, as to some Internet paths suffering from serious reordering, knowledge of both time lag threshold and packet lag threshold can help improve TCP performance by distinguishing reordering and loss precisely.

## 5 Reordering and Network Topology

There are mainly two approaches to deal with packet reordering in the Internet. The one is to improve the TCP on end-hosts, making TCP more robust to reordering. The other is to improve the routers in the Internet which is the main cause of reordering. In this section, we discuss the approach to infer reordering-generating spots in the Internet. It is a prerequisite for the latter topic, on which little research is published.

A packet often goes through many routers from the sender to the receiver. When a packet arriving at the receiver reordered, we can not find out the reordering-generating spot without further information. However, if a router is a reordering-generating spot, all the packets transmitted by it may be reordered in theory. In our measurement, all the forward-paths from remote web sites to the local measurement host form a tree, in which the measurement host is the root, the routers are the middle nodes and the remote web sites are the leaves. If a router in the tree generates reordering, all its leaves may be affected. So we can infer

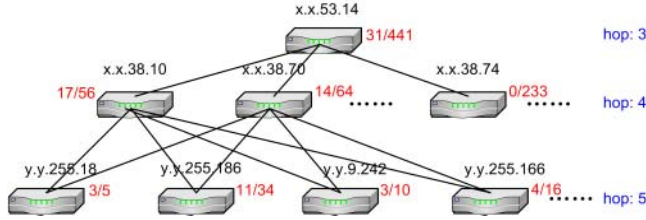


Fig. 6. Inferring the reorder-generating spots in the routing tree (Condition 1)

reorder-generating spot by studying the pertinence of leaves found as Reorder Sites. We use *traceroute* to gather the route information from the measurement host to all the 10,647 web sites. Then we generate a backward-path route tree in which the measurement host is also the root. Since the route architecture of CERNET is mainly symmetric, forward-path tree and backward-path tree within it could be approximately treated as the same. We introduce a metric  $R_r$  (reorder ratio) to every router as the main parameter for the reorder-generating spot judgment, which is defined by (1):

$$R_r = \frac{R}{T} \tag{1}$$

where,  $R$  is the number of Reorder Sites that go through the router and  $T$  is the total number of sites that go through it.

If a router has one of the below two characters, it is probably a reorder-generating spot in the network:

- The router’s  $R_r$  is by far higher than its previous-hop’s and other routers’ of the same hop. Its next-hop routers also have got high and close  $R_r$ .
- All of the router’s previous-hop routers have got high  $R_r$ , and its next-hop routers also get high  $R_r$ .

As Fig. 6 shows, the  $R_r$  of the 4th hop routers x.x.38.10 and x.x.38.70 are obviously higher than the 3rd hop x.x.53.14. Firstly, since none of the 233 sites which goes through the 4th hop router x.x.38.74 experienced reordering, it is impossible that the reordering-generating spot locates at the 3rd hop or its previous ones. Moreover, the routers of 5th hop have got high and close  $R_r$ . So the two IP of 4th hop are probably the same router with “local parallelism”. In fact, x.x.38.10 and x.x.38.70 are the same equipment in CERNET Super Computer Center with two gigabit paths connected to the 3rd router x.x.53.14. The several 5th hop routers are all star connected to a GSR (Gigabit Switch Router). So we come to the conclusion that the reorder-generating spot in Fig. 6 locates between the 3rd router x.x.53.14 and the 4th hop routers with IP x.x.38.10 and x.x.38.70.

As Fig. 7 shows, the 4 routers of 8th hop and the 6 routers of 10th hop which connects to the two 9th hop routers have got high and close  $R_r$ <sup>5</sup>. Since

<sup>5</sup> The 10th hop router with IP: y.y.139.214 is an exception, which contains too little data and could be neglected.

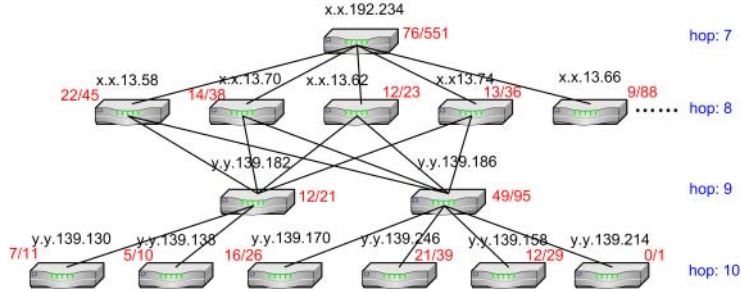


Fig. 7. Inferring the reorder-generating spots in the routing tree (Condition 2)

the  $R_r(0.102)$  of 8th hop router 202.96.13.66 is much lower than the other four  $R_r$  (with an mean of 0.435), routers previous to hop 7 can not be the reorder-generating spot. On the other hand, all the 6 routers of 10th hop are connected to the 9th routers with a single path. It is very unlikely that they all generate reordering at the same time and result in the relationship of in Fig. 7. So it is most probably that the two IP close to each other belongs to the same router with “local parallelism”. And the 8th hop and 9th hop routers along with the multi-paths between them are probably the reorder-generating spot in Fig. 7. Although the routers in Fig. 7 are not in CERNET, the approach above has general significance as long as we know the forward-path route tree. It might not exactly locate where the reorder-generating spot is, but it does can help us a lot to exclude reduce the scope. Further study may introduce multi-point observation to gather more route information.

## 6 Conclusions

This paper provides an insight into both the attributes of packet reordering in the Internet itself, and the relationship between reordering and network topology. Firstly, a measurement methodology with a single-point reorder-judging algorithm is proposed, which is suitable for the above two purposes. Then, measurement results of 208 thousand connections with totally 3.3 million data packets are presented, in which about 3.2% of all the packets are observed reordered. We find that reordering is not prevalent in the entire Internet but significantly site-dependent. We also note that certain threshold can be found to effectively help distinguish reordering and loss on some heavily reordering paths. Nevertheless, the distribution of packet lag of reordering and retransmission implies that in most cases reordering will not trigger the fast retransmission algorithm, thus will not affect the TCP performance seriously. Moreover, we propose a novel and relatively reliable approach to infer reorder-generating spots in the Internet by studying the pertinence of reordering sites and the routing tree. It is a first step of our work to analyze the relationship between packet reordering and network topology. Currently, deployment of multi-point reordering measurement

with more precisely data crawling is considered to overcome the limitations of single-point observation at the TCP receiver, such as the exact forward-path route tree is unknown and the possible difficulty of inferring reorder-generating should there be a reorder-generating spot very close to the root of the route tree.

## References

- [1] Paxson, V.: End-to-end Internet Packet Dynamics. *IEEE/ACM Transactions on Networking*, Vol. 7, Issue 3. (1999) 277–292 [351](#), [352](#)
- [2] Jaiswal, S., Iannaccone, G., Diot, C., et al.: Measurement and Classification of Out-of-Sequence Packets in a Tier-1 IP Backbone. Sprint ATL Technical Report TR02-ATL-071121 (2002) [351](#), [352](#), [353](#)
- [3] Bennett, J. C. R., Partridge, C., Shectman, N.: Packet Reordering is Not Pathological Network Behavior. *IEEE/ACM Transaction on Networking*, Vol. 7, Issue 6. (1999) 789–798 [351](#), [352](#)
- [4] Bellardo, J., Savage, S.: Measuring Packet Reordering. Department of Computer Science and Engineering, University of California at San Diego (2002) [352](#)
- [5] Liu, H.: A Trace Driven Study of Packet Level Parallelism. Proc. International Conference on Communications (ICC), New York, NY (2002) [352](#)
- [6] Lu, G. H., Li, X.: On the Correspondency between TCP Acknowledgment Packet and Data Packet. Proc. Internet Measurement Conference (IMC), Miami Beach, FL, USA (2003) 285–294 [353](#)
- [7] Iannaccone, J., Jaiswal, S., Diot, C.: Packet reordering inside the Sprint backbone. Sprint ATL Technical Report TR01-ATL-062917 (2001) [356](#)
- [8] Floyd, S., Henderson, J.: The NewReno Modification to TCP’s Fast Recovery Algorithm. RFC 2582 (1999) [352](#)
- [9] Floyd, S., Mahdavi, J., Mathis, M., et al.: An Extension to the Selective Acknowledgement (SACK) Option for TCP. RFC 2883 (2000) [352](#)
- [10] Zhang, M., Karp, B., Floyd, S., Peterson, L.: Improving TCP’s Performance under Reordering with DSACK. Technical Report TR-02-006, International Computer Science Institute (2002) [352](#)
- [11] Blanton, E., Allman, M.: On Making TCP More Robust to Packet Reordering. *ACM Computer Communication Review*, Vol. 32, Issue 1. (2002) [352](#)
- [12] Stevens, W.: TCP Slow Start, Congestion Avoidance, Fast Retransmit, and Fast Recovery Algorithms. RFC 2001 (1997) [350](#)
- [13] McCreary, S., Claffy, K.: Trends in Wide Area IP Traffic Patterns: A View from Ames Internet Exchange. <http://www.caida.org/outreach/papers/2000/AIX0005/> [352](#)
- [14] Laor, M., Gendel, L.: The Effect of Packet Reordering in a Backbone Link on Application Throughput. *IEEE Network*, Vol. 16, Issue 5. (2002) [351](#)