Relative Delay Estimator for Multipath Transport

Fei Song
Beijing Jiaotong University
Beijing, China

Hongke Zhang
Beijing Jiaotong University
Beijing, China

Sidong Zhang
Beijing Jiaotong University
Beijing, China

Fernando M.V. Ramos
University of Cambridge
Cambridge, UK

Jon Crowcroft
University of Cambridge
Cambridge, UK

ABSTRACT

Due to the dynamic nature of the Internet, the characteristic parameters of network paths are continually changing. The round trip time (RTT) can be used to estimate retransmission timeouts with reasonable accuracy. However, using RTT to evaluate forward or backward delays is not suitable. By identifying this shortcoming, we propose a relative delay estimator (RDE) to make a distinction between all available paths and build a retransmission policy based on it.

Categories and Subject Descriptors

C.2.2 [**Computer Systems Organization**]: Computer Communication Networks—*Network Protocols*

General Terms

Algorithms, Design, Performance

1. INTRODUCTION

Nowadays, most laptops and smart phones have several interfaces to access the Internet: WiFi, 3G, etc. It is natural to consider using them simultaneously to improve performance. For that to happen, the transport layer protocols need to be redesigned to meet the requirements of transmitting using multiple paths. Recent work has proposed several different schemes to increase throughput of multipath transport, by choosing the best path for retransmission based on window size, ssthresh, loss rate, etc. [2].

Understanding forward or backward delays separately may help us in designing more efficient mechanisms for multipath transport. For instance, voice applications are very sensitive to delay, thus they could benefit from schemes that take delay into account. We propose a Relative Delay Estimator (RDE) to compare the relative one way delay of different paths without clock synchronisation. As an initial application of RDE, we also present a novel retransmission policy.

2. BASIC CONCEPTS OF RDE

The RTT can be separated into four parts:

$$RTT = t_{Fp} + t_{Fq} + t_{Bp} + t_{Bq} \quad (1)$$

CoNEXT Student Workshop'09, December 1, 2009, Rome, Italy.
ACM 978-1-60558-751-6/09/12.

where t_{Fp} and t_{Fq} are the propagation delay and queueing delay in the forward path, respectively. t_{Bp} and t_{Bq} are the same quantities but in the backward path.

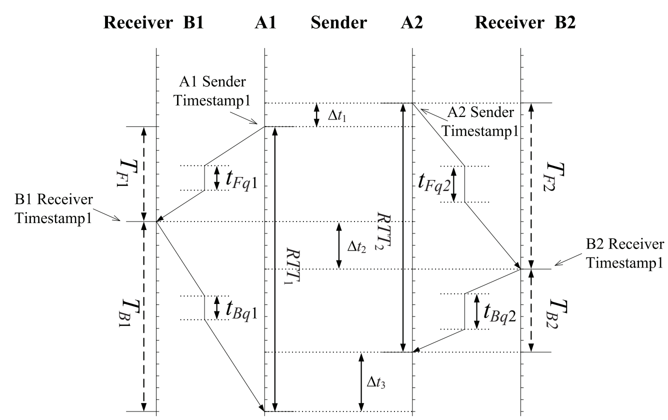

Figure 1: Two paths scenario with queue delay

If all entities in a network were synchronised, calculating the one way delay would be straightforward. However, this is not true in the current Internet. The Network Time Protocol (NTP) can indeed be used for this purpose, but since NTP is not always available in all end hosts, our proposal is not to calculate the absolute delays, but the **relative** one way delays of each path. This information will be enough for selecting the best path in terms of delay.

Consider the simple two paths transport scenario in Figure 1. One sender and one receiver are represented, each having two network interfaces. $B1$ and $B2$, for instance, are the two network interfaces of the receiver B. The double arrows with solid (such as: $\Delta t1$, $\Delta t2$) or dash lines (such as: T_{F1}, T_{F2}) indicate whether or not we can calculate these values based on timestamps. $\Delta t_x(x = 1, 2, 3)$ represent the differences between corresponding path1's timestamp and path2's timestamp. According to the relationship of variables mentioned before, we obtain the system of linear equations:

$$\begin{array}{ll} T_{F1} + T_{B1} = RTT_1 & T_{F2} + T_{B2} = RTT_2 \\ T_{F1} - T_{F2} = \Delta t_2 - \Delta t_1 & T_{B1} - T_{B2} = \Delta t_3 - \Delta t_2, \end{array} \quad (2)$$

This system of linear equations has infinite solutions. Nevertheless, we have enough information to calculate the relative delays between two paths (i.e., $T_{F1} - T_{F2}$ and $T_{B1} - T_{B2}$) and get the path with minimal forward or backward delay.

3. RETRANSMISSION POLICY

We propose two schemes, one for fast retransmission, the other for timeouts. Fast retransmission usually occurs with sporadic packet losses, in a low-congestion situation. In this case, the retransmitted packet and its ACK will be sent on the path with minimum forward and backward delay, respectively. On the other hand, a timeout generally signals heavy congestion or path failure. In this case, the paths where the timeout occurred are marked as "potentially-failed", or PF (check [3] for details). A PF path will not be used for forwarding any packets, except a heart beat packet to test the path condition. After removing the PF paths from the list of available paths, the sender will select paths for the retransmitted packet and its ACK as before.

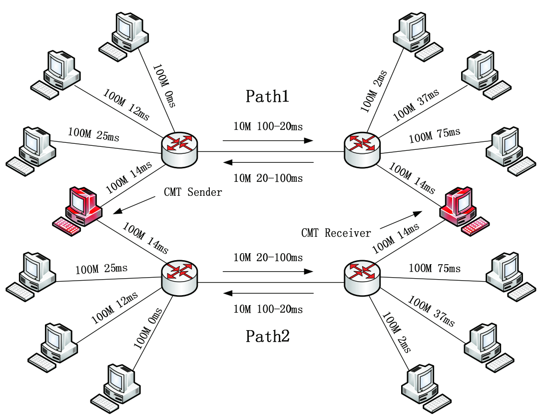

Figure 2: Topology

4. EVALUATION

We implemented RDE and retransmission policy into NS2-2.33 CMT-SCTP module and use the topology shown in Figure 2 for evaluation. Eight different scenarios were tested for two experiments explained below, by varying the forward delay, backward delay and the loss rate. The RTT on both paths is set to $120ms$. The forward delay on path 2, FD_2, is always equal to $120ms - FD_1$. Due to space constraints, only one scenario for each experiment is presented here. We have run the simulation using 20 random seeds. The figures illustrate the average throughput with error bars representing maximum and minimum values for each case. All other results and more detailed discussions are available online[1].

Experiment 1: In Figure 3 we compare RDE (without PF marking) with the original five policies presented in [2] on the scenarios with no path failure. Our policy has better performance when the forward or backward delays are different between the available paths. At the moment we are doing some experiments to prove that such difference is very common in the current Internet.

Experiment 2: In Figure 4, the performance of RDE and PF-RDE (with PF marking) are compared. If there is no path failure, the results are very similar. However, when a path fails, PF-RDE can detect it earlier and select another path for retransmission, thus improving the throughput.

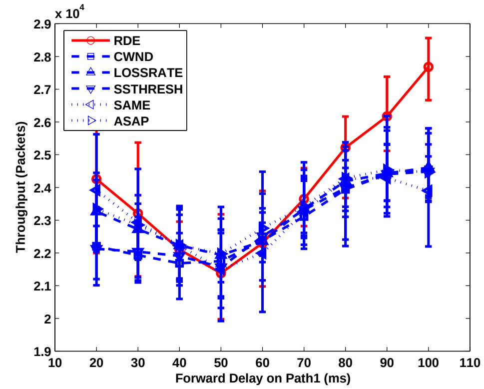

Figure 3: Results of experiment 1

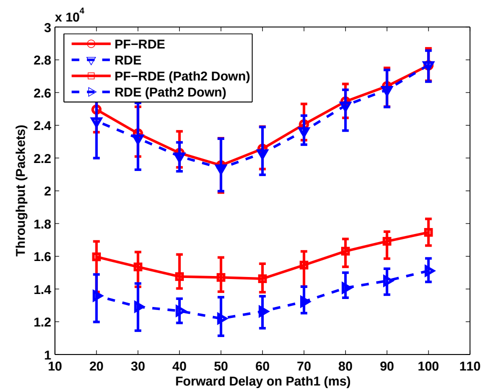

Figure 4: Results of experiment 2

5. CONCLUSIONS AND FUTURE WORK

We presented a relative delay estimator for multipath transport. As an initial application, a retransmission policy is proposed and evaluated in different scenarios. Simulation results show that the throughput can be improved when the one way delay between two paths are different.

As future work we intend to use RDE's timing information for bandwidth estimation and for shared bottleneck detection. More details on this future work can be found in [1].

6. REFERENCES

[1] F. Song et al. An estimator of forward and backward delay for multipath transport. Computer Laboratory, University of Cambridge, UCAM-CL-TR-747, 2008.

[2] J. Iyengar et al. Concurrent multipath transfer using sctp multihoming over independent end-to-end paths. In *IEEE/ACM Transaction on Networking*, 2006.

[3] P. Natarajan et al. Concurrent multipath transfer using transport layer multihoming: Introducing the potentially-failed destination state. In *IFIP Networking*, 2008.

[1]http://www.cl.cam.ac.uk/~fs332/RDE.pdf