An Adaptive Transport Protocol for Multimedia Communication

Dane Dwyer, Sungwon Ha, Jia-Ru Li, Vaduvur Bharghavan
University of Illinois at Urbana-Champaign

IEEE International Conference on Multimedia Computing and Systems, 1998

Presented by Hui-Wen Kan   2006/09/07

Outline

- Introduction
- HPF
  - Window management
  - Application interface
- Measurements
- Conclusions

Introduction

- Motivation
  - User traffic become increasingly multimedia oriented and *heterogeneous*
  - Neither of TCP and UDP is well suited for multimedia flows

Introduction (cont.)

- Goal
  - Support multiple interleaved reliable and unreliable data sub-streams
  - Decouple the congestion control and reliability mechanisms
  - Use application-defined priorities for the link scheduler to drop low-priority packets during congestion

HPF

- HPF (Heterogeneous Packet Flow)
  - Window management
  - Application interface

Window management

- Similar with TCP
  - Use or enhance the mechanisms that are provided by TCP for flow control, reliability, and sequencing
- Difference
  - Congestion control mechanism
  - Support interleaved reliable and unreliable packet sub-streams

Window management (cont.)

- Flow control, reliability, and sequencing
  - Packet may have been either a low or a high priority packet
    - low priority: the dropped packet be ignored
    - high priority: should be retransmitted
  - Each packet needs to identify the previous high priority packet (called *prevHIGH*)

Window management (cont.)

- Example

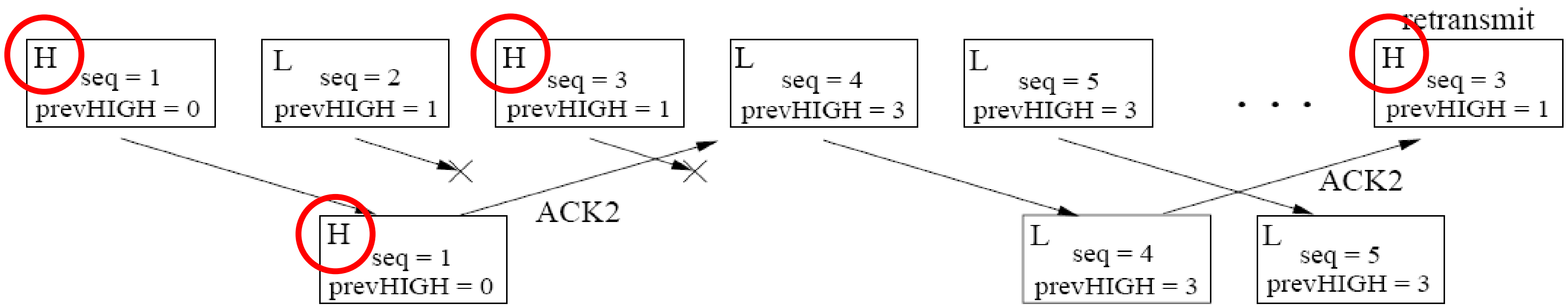


Figure 1. Example of acknowledgements in HPF. Only high priority packets that are lost will be retransmitted. When the receiver gets the retransmitted high priority packet with seq=3, it will send ACK6 (rather than ACK2). The lost low priority packet with seq=2 is implicitly acknowledged and will not be retransmitted.

Window management (cont.)

- Congestion control
  - The sender estimates the congestion window based on the fraction of received packets in the current window

Window management (cont.)

- Compute the fraction of received packets
  - Use a congestion estimation window
  - The sender maintains ***epoch*** and ***cwnd***
  - The receiver maintains ***sender.epoch*** and ***sender.cwnd***
  - Each packet has two fields ***packet.epoch*** and ***packet.cwnd***

Window management (cont.)

- if packet.epoch == sender.epoch
  - received packet counter ++
- if packet.epoch < sender.epoch
  - the congestion information is ignored
- if packet.epoch > sender.epoch
  - the receiver starts a new estimation window
  - received packet counter = 0
  - sender.epoch = packet.epoch
  - sender.cwnd = packet.cwnd

Window management (cont.)

- Updating the congestion window
  - if cong_ack.fraction > α

    cwnd = f_I (cwnd)
  - if cong_ack.fraction <= α

    cwnd = f_D (ack.cwnd, ack.fraction)
  - if a timeout occurs

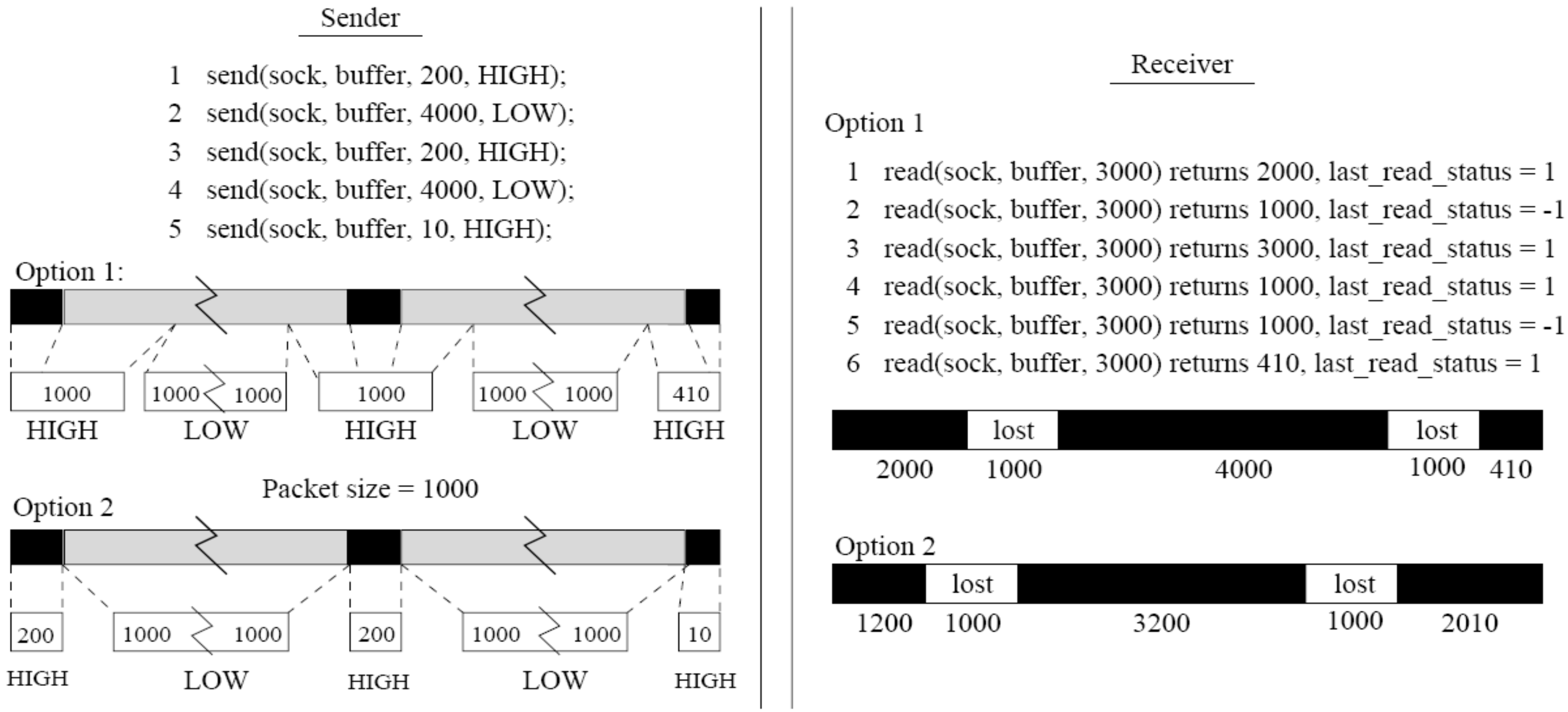
    ssthresh = cwnd/2

α : (1- random packet loss probability)

Application interface

- The application must to signal the priority level
  - Maximize throughput
    - Merge reliable and unreliable data into a packet
  - Maximize adaptation
    - Merge only ‘like-priority’ data bytes
  - Loss based
    - Based on the fraction of received packets

Application interface (cont.)



Measurements

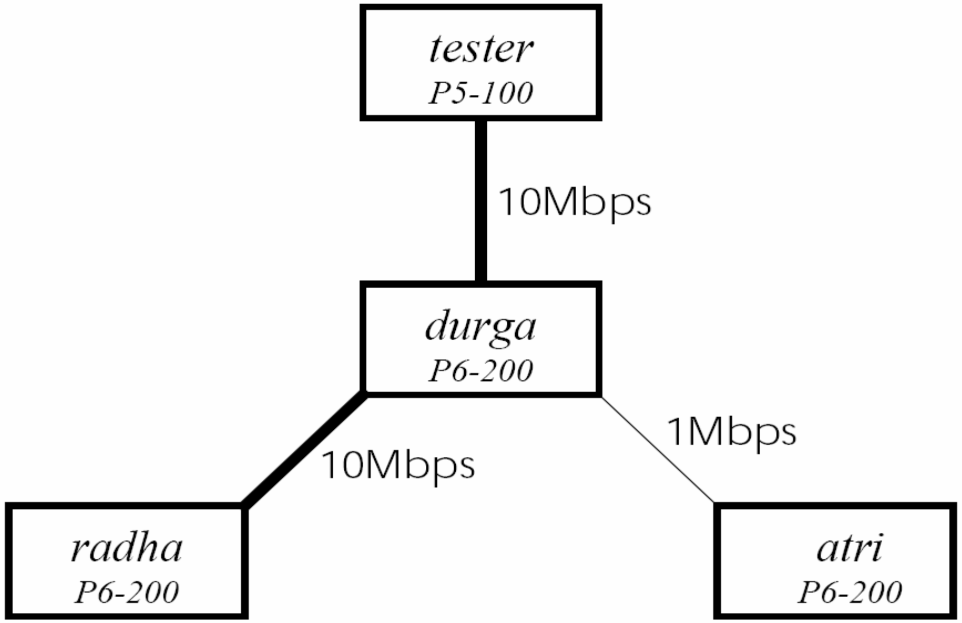


Figure 3. The experimental testbed configuration used for the performance tests.

Measurements (cont.)

Protocol	High:Low Ratio	Packets Dropped	Improvement vs. TCP
TCP	—	0%	—
HPF	1:0	0%	-4.0%
	7:3	1.2%	2.2%
	5:5	5.1%	16.7%
	3:7	5.5%	21.3%

Table 1. The performance of HPF vs TCP at various priority ratios and bursty UDP traffic

Measurements (cont.)

Protocol	High:Low Ratio	Packets Dropped	Improvement vs TCP
TCP	—	0%	—
HPF	1:0	0%	-4.3%
	7:3	1.1%	8.0%
	5:5	5.3%	29.7%
	3:7	5.7%	32.5%

Table 2. The performance of HPF vs TCP at various priority ratios with multiple concurrent streams

Conclusions

- Support multiple interleaved reliable and unreliable data sub-streams
- Decouple the congestion control and reliability mechanisms
- Use application-defined priorities for the link scheduler to drop low-priority packets during congestion