CS 168 Introduction to the Internet
Fall 2017 Scott Shenker

Discussion 11: Congestion Control

1 True False

1. UDP uses congestion control.
2. Flow control slows down the sender when the network is congested.
3. For TCP timer implementations, every time the sender receives an ACK for a previously unACKed packet, it will recalculate ETO.
4. CWND (congestion window) is usually smaller than RWND (receiver window).
5. AIMD is the only ”fair” option among MIMD, AIAD, MIAD, and AIMD.

2 Impact of Fast Recovery

Consider a TCP connection, which is currently in Congestion Avoidance (AIMD).

- The last ACK sequence number was 101.
- The CWND size is 10 (in packets).
- The packets #101-110 were sent at t=0,0.1,...,0.9 (sec), respectively.
- The packet #102 is lost only for its first transmission.
- RTT is 1 second.

Fill in the tables below, until the sender transmits the packet #116.

1. Without fast recovery:
   - On new ACK, $CWND += \frac{1}{\lfloor CWND \rfloor}$
   - On triple dupACKs, $SSTHRESH = \left\lfloor \frac{CWND}{2} \right\rfloor$, then $CWND = SSTHRESH$.

Time (sec)	Receive ACK (due to)	CWND	Transmit Seq # (mark retransmits)
1.0	102 (101)	$10 + \frac{1}{10} = 10.1$	111
1.2	102 (103)	10.1	/
1.3	102 (104)	10.1	/

2. With fast recovery:

   - On triple dupACKs, $SSTHRESH = \left\lfloor \frac{CWND}{2} \right\rfloor$, then $CWND = SSTHRESH + 3$, enter fast recovery.
   - In fast recovery, $CWND += 1$ on every dupACK.
   - On new ACK, exit fast recovery, $CWND = SSTHRESH$

Time (sec)	Receive ACK (due to)	CWND	Transmit Seq # (mark retransmits)
1.0	102 (101)	$10 + \frac{1}{10} = 10.1$	111
1.2	102 (103)	10.1	/
1.3	102 (104)	10.1	/

3 AIMD Generalization and Derivation

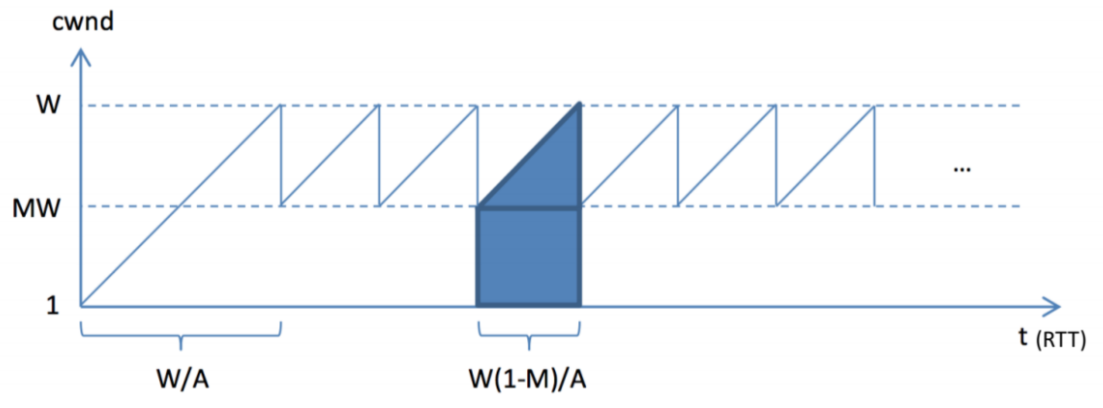

Figure 1: Graph of *Window size vs. time* referenced in AIMD Generalization and Derivation

Consider a generalized version of AIMD, where:

- For every window of data ACKed, the window size increases by a constant A
- When the window size reaches W, a loss occurs, and the window size is multiplied by a constant $M < 1$

For simplicity, *assume that* $W(1-M)$ *is divisible by* A. Thus, the window sizes will cycle through the following: WM, $WM + A$, $WM + 2A$, ... W. Let the RTT to denote the packet round trip time. A graph of window size versus time is referenced in Figure 3.

We present the following questions.

1.) What is the average throughput? As we did in the lecture slides, express in your answers in the number of packets, so we do not need to consider MSS.

2.) Calculate the loss probability p, using W and M.

3.) Derive the formula for throughput in part 1 when $M = 0.5$ and $A = 1$ and try using only p and RTT.