Agenda for Today

• Reminders
• Background on congestion control in datacenters
• Homa discussion
• Swift discussion
Reminders

• Warm-up assignment
  • Due today at 11:59 pm
• Projects
  • See notes on Canvas
  • Proposals due on 10/20
  • Talk to us if you want help brainstorming ideas
• For Thursday:
  • No need to review the “Killer Microseconds” paper
  • Do submit a review for Shenango
Congestion Control in Datacenters
TCP

- Congestion window (cwnd): number of bytes that can be outstanding at once
- TCP adjusts the cwnd based on additive increase/multiplicative decrease (AIMD)
  - No congestion: cwnd += 1
  - Congestion: cwnd = cwnd / 2
- Detect congestion when a packet is dropped

“saw-tooth” pattern

• TCP does not work well in datacenters
  • Large “background” flows cause queueing in the network
  • Latency-sensitive “foreground” traffic suffers from high latency
• Particularly bad with partition/aggregate workloads
  • Applications need low tail latency (e.g., 99.9%)
• Goal: decrease the sending rate before the queues fill up
  • Mark packets when queueing exceeds a threshold

What is optimal?

- Goal: minimize the average time to send a message
- Optimal policy: shortest remaining processing time (SRPT)
  - Sends the message with the fewest bytes remaining first
- Challenges with SRPT
  - Need to know the message size
  - May starve long messages
  - Not necessarily optimal with multiple switches
- Many protocols approximate SRPT
  - pFabric, PIAS, pHost, Homa
Homa Discussion
Swift Discussion