CS 204: Scheduling

Jiasi Chen

Lectures: MWF 12:10-1pm in WCH 139

http://www.cs.ucr.edu/~jiasi/teaching/cs204_spring16/

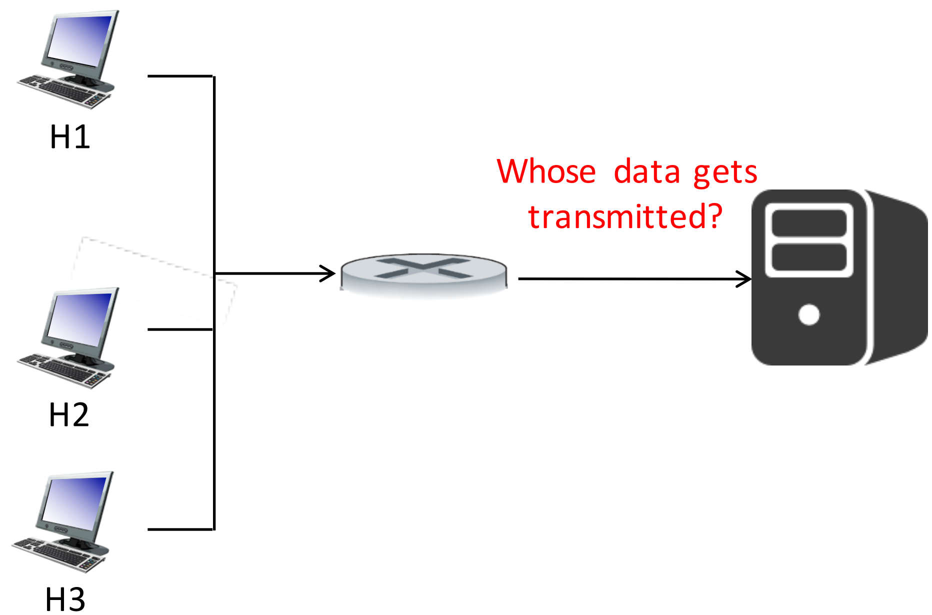
Overview

- What is scheduling?
  - Round robin
  - Weighted round robin
- Generalized processor sharing (GPS)
- Implementations of GPS
  - Deficit round robin
  - Weighted fair queuing
- Rate control with GPS: Leaky bucket
- Paper discussion

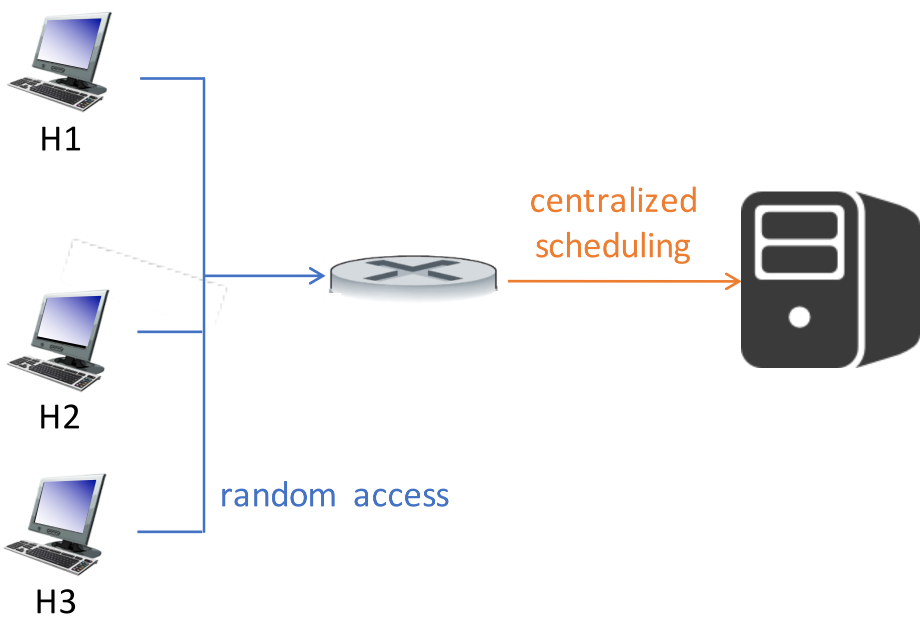
Q: How should a common resource be shared between multiple users?

What is scheduling?

- Send packets from multiple hosts
- Required features
  - Fair
  - Easy to implement
  - Scalable
  - Work-conserving
- Desirable features
  - Admission control
  - QoS for different types of traffic

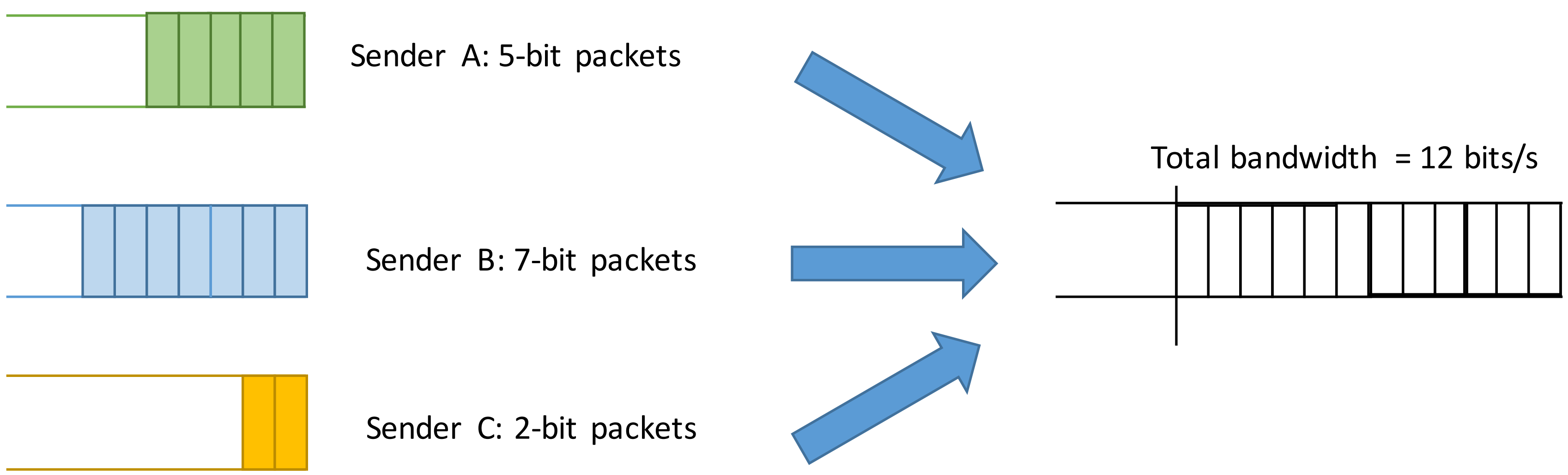


What is NOT scheduling?



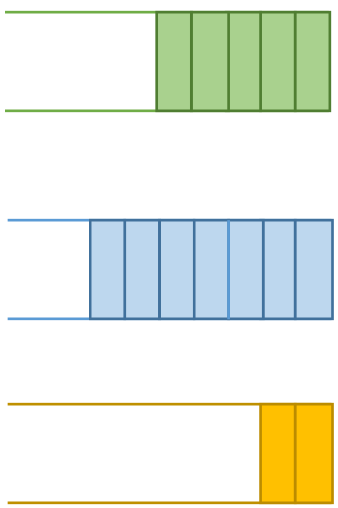
Uplink?
Downlink?

Toy example


Sender A: 5-bit packets
Sender B: 7-bit packets
Sender C: 2-bit packets
Total bandwidth = 12 bits/s

Fairness

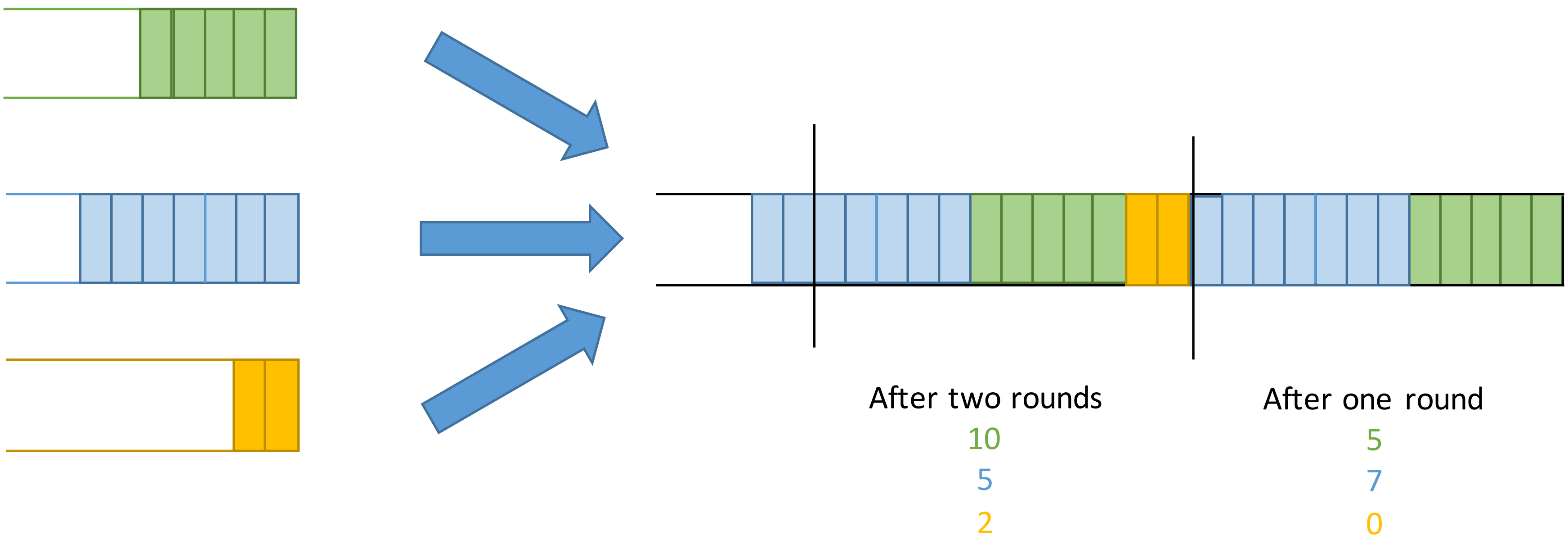
- How much of the link does each user get?
- One measure of fairness: max-min fair
  - Maximize the minimum rate across all users



	Max-min fair	NOT max-min fair
Sender A: 5-bit packets	5	4
Sender B: 7-bit packets	5	6
Sender C: 2-bit packets	2	2

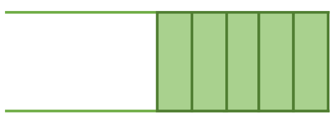
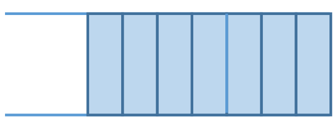
Round robin

- Choose 1 packet from each subqueue



Weighted round robin

- Choose certain # packets from each subqueue
  - # of packets sent = weight / packet size

	Weights	# packets sent	Number of bits sent
	5	5/5 = 1 → 7	7 packets * 5 bits/packet = 35 bits
	5	5/7 → 5	5 packets * 7 bits/packet = 35 bits
	2	2/2 = 1 → 7	7 packets * 2 bits/packet = 14 bits

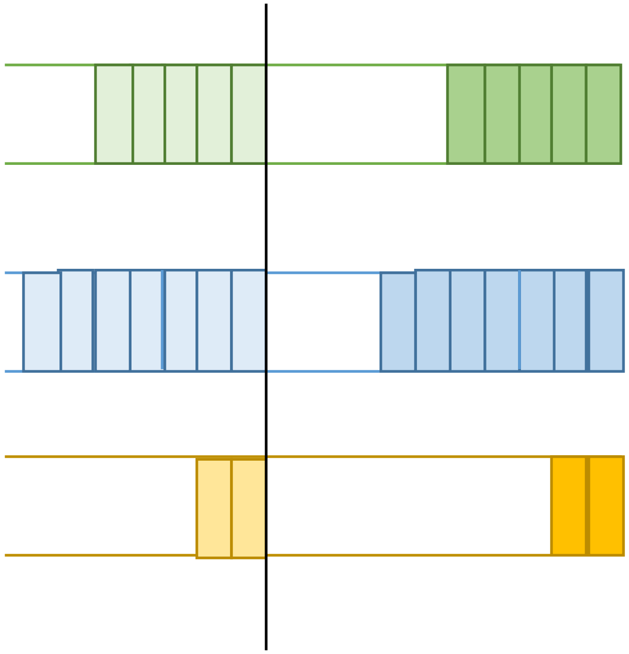
Overview

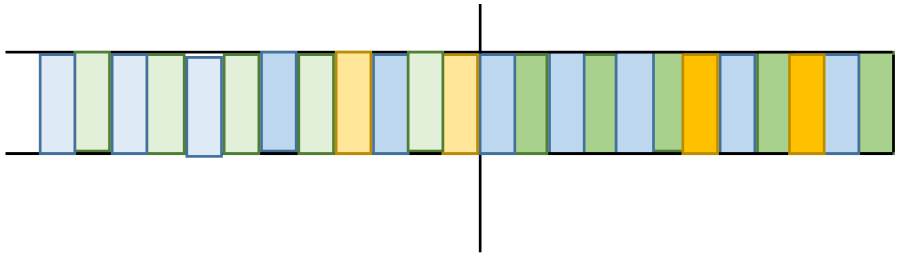
- What is scheduling?
  - Round robin
  - Weighted round robin
- Generalized processor sharing (GPS)
- Implementations of GPS
  - Deficit round robin
  - Weighted fair queuing
- Rate control with GPS: Leaky bucket
- Paper discussion

Q: How should a common resource be shared between multiple users?

Bit-by-bit Round Robin

- Not implementable

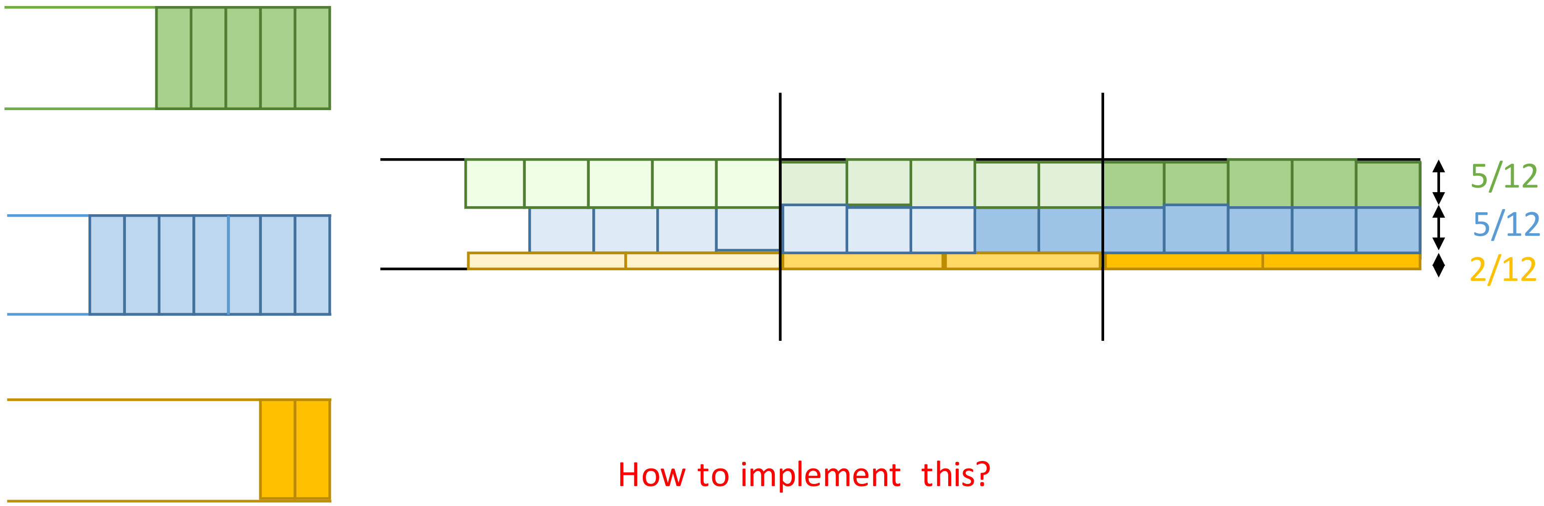




After two rounds	After one round
10	5
10	5
4	2

Generalized processor sharing

- Send infinitesimal amount of bits each round
- Idealized version of bit-by-bit round robin
- Not implementable, but achieves perfect fairness



How to implement this?

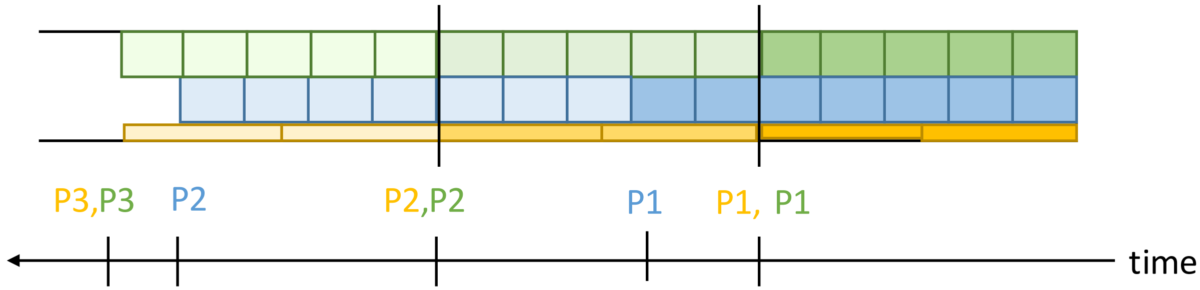
Overview

- What is scheduling?
  - FIFO
  - Round robin
  - Weighted round robin
- Generalized processor sharing (GPS)
- Implementations of GPS
  - Deficit round robin
  - Weighted fair queuing
- Rate control with GPS: Leaky bucket
- Paper discussion

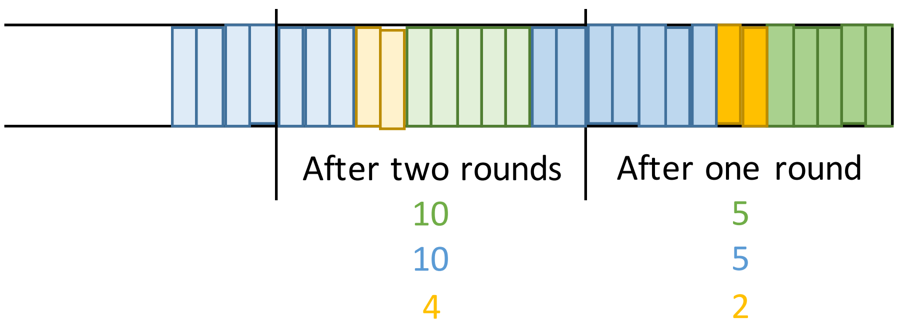
Q: How should a common resource be shared between multiple users?

Weighted fair queuing

- See which packets would finish first under GPS



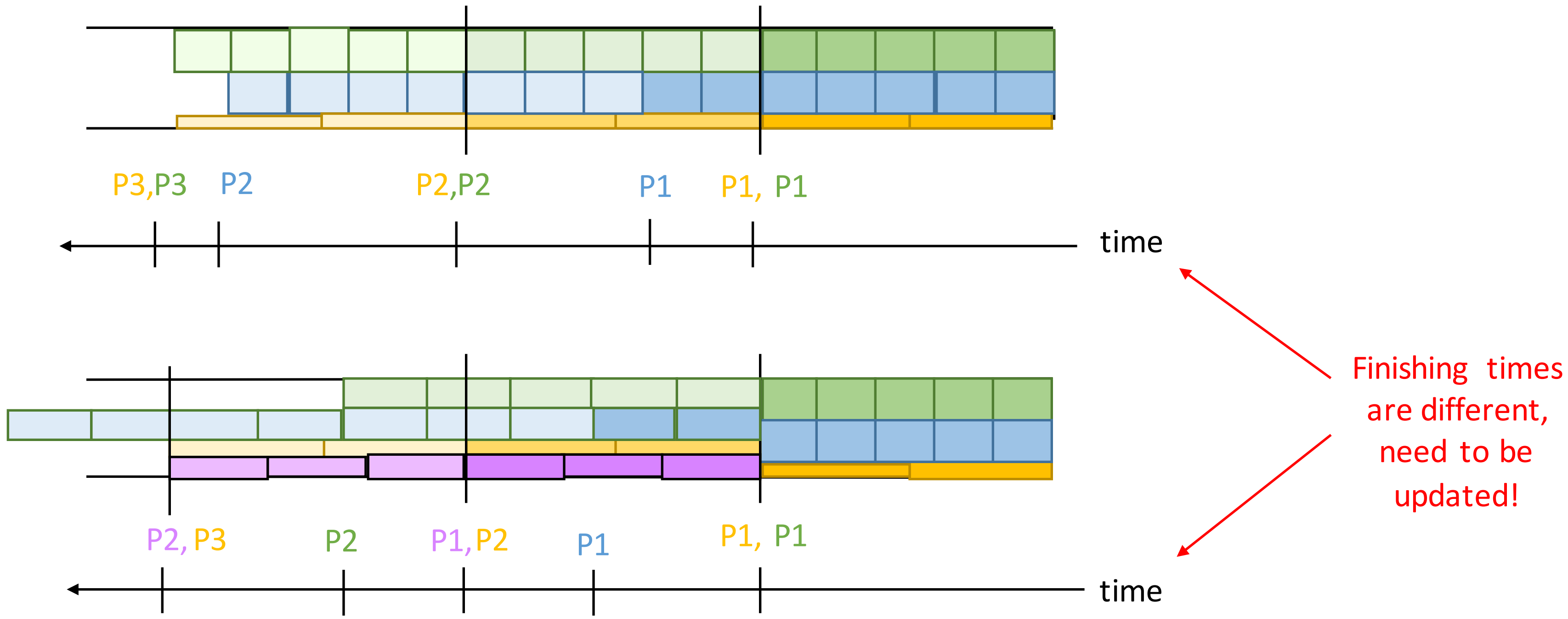
- Send packets in that order



O(log(n)) complexity
Need to sort the list of finishing times

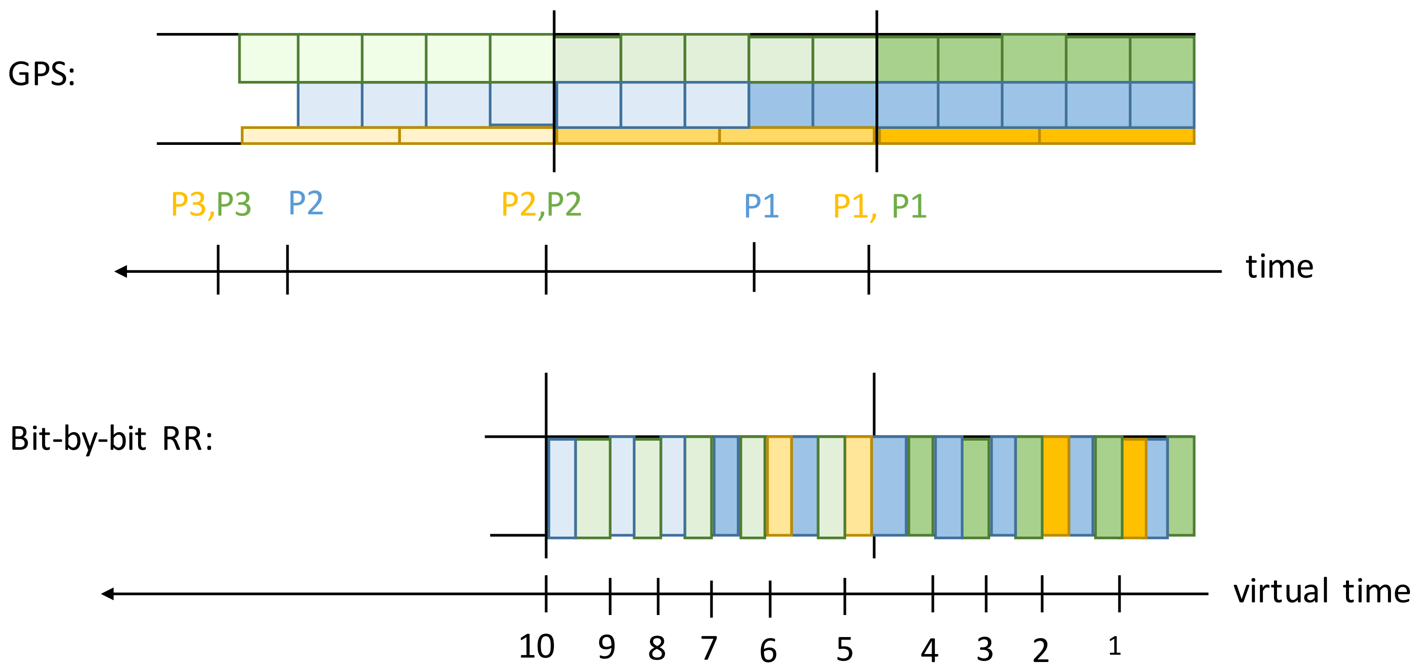
Weighted fair queuing (2)

- Problem: what if new user enters the system?



Weighted fair queuing (3)

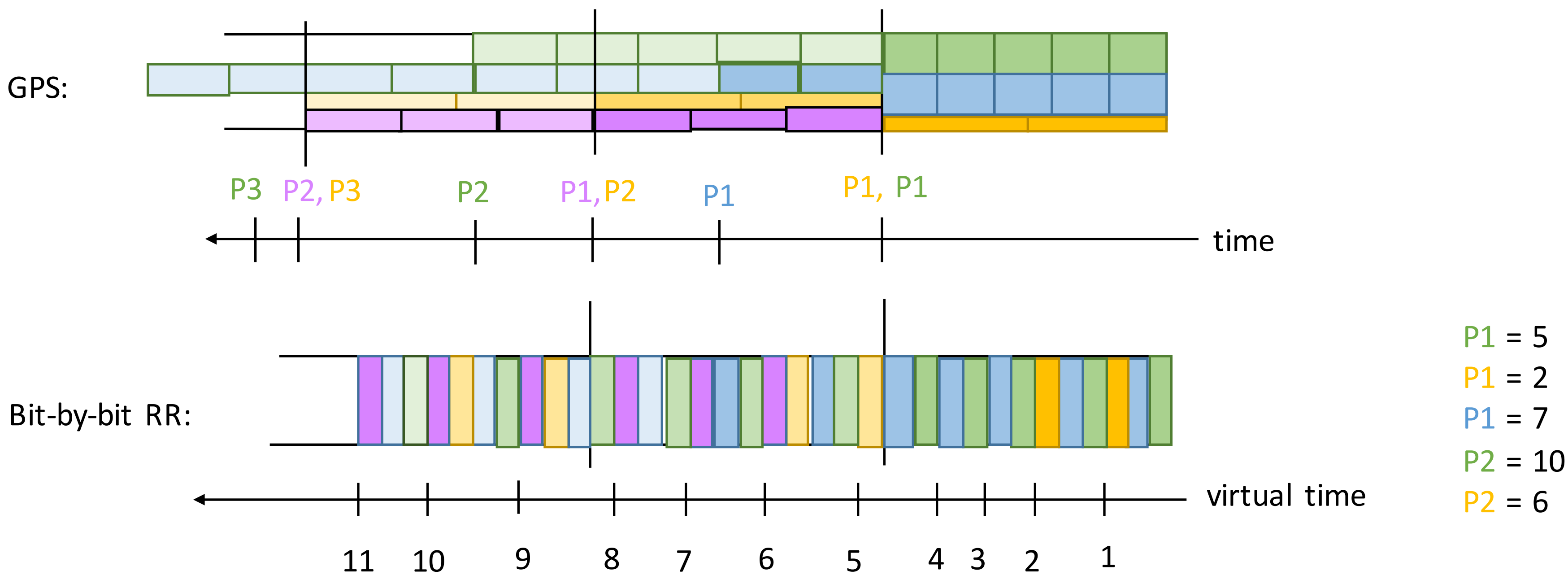
- Solution: use “virtual time” = # of bit-by-bit round robin rounds



P1 = 5
P1 = 2
P1 = 7
P2 = 10
P2 = 6

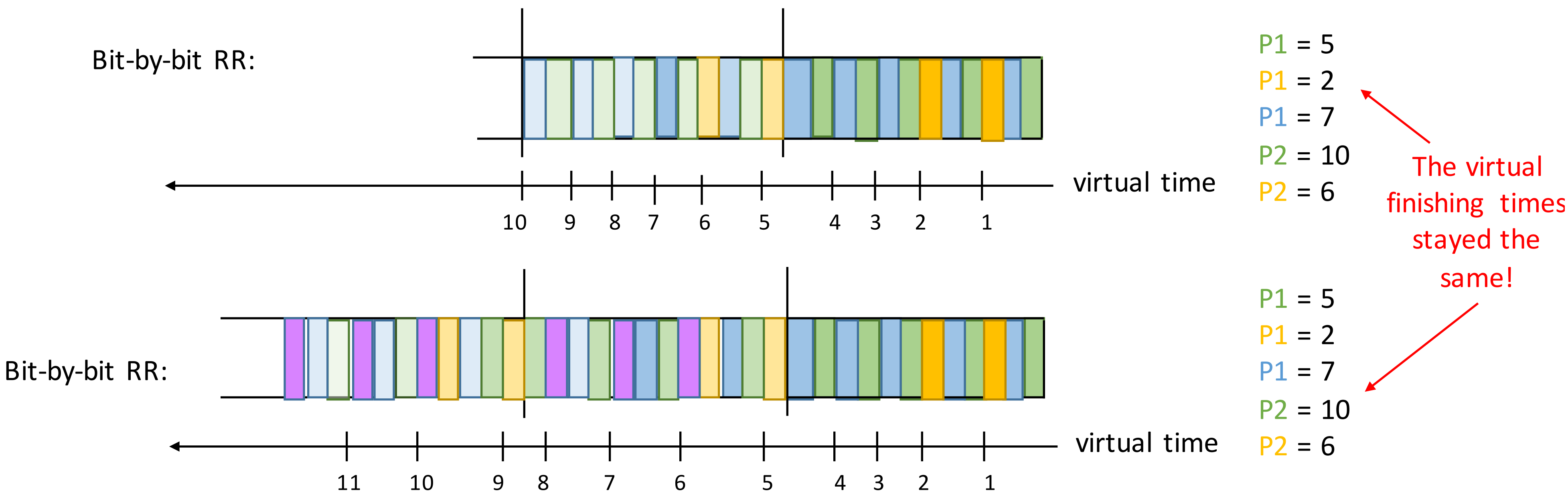
Weighted fair queuing (4)

- Let’s calculate the virtual finishing time with the new user



Weighted fair queuing (5)

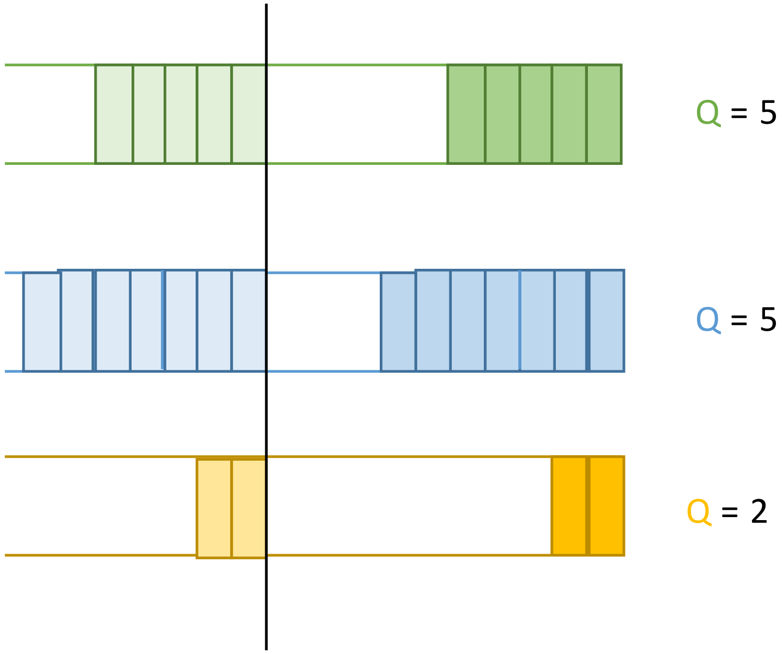
- Even if new user joins, the "virtual time" stays the same



Note: BBRR virtual time is shown here for ease of exposition; WFQ virtual time is slightly different. See Parekh-Gallager for details.

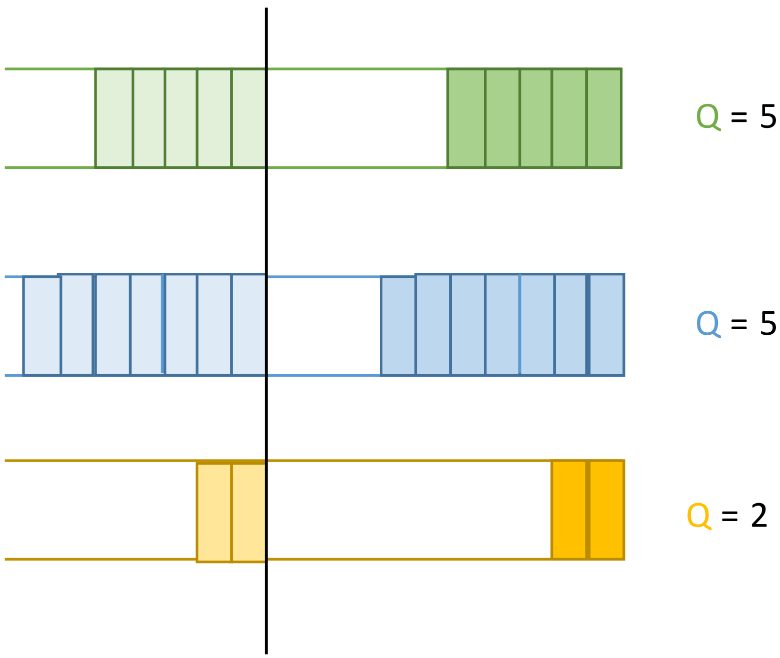
Deficit round robin

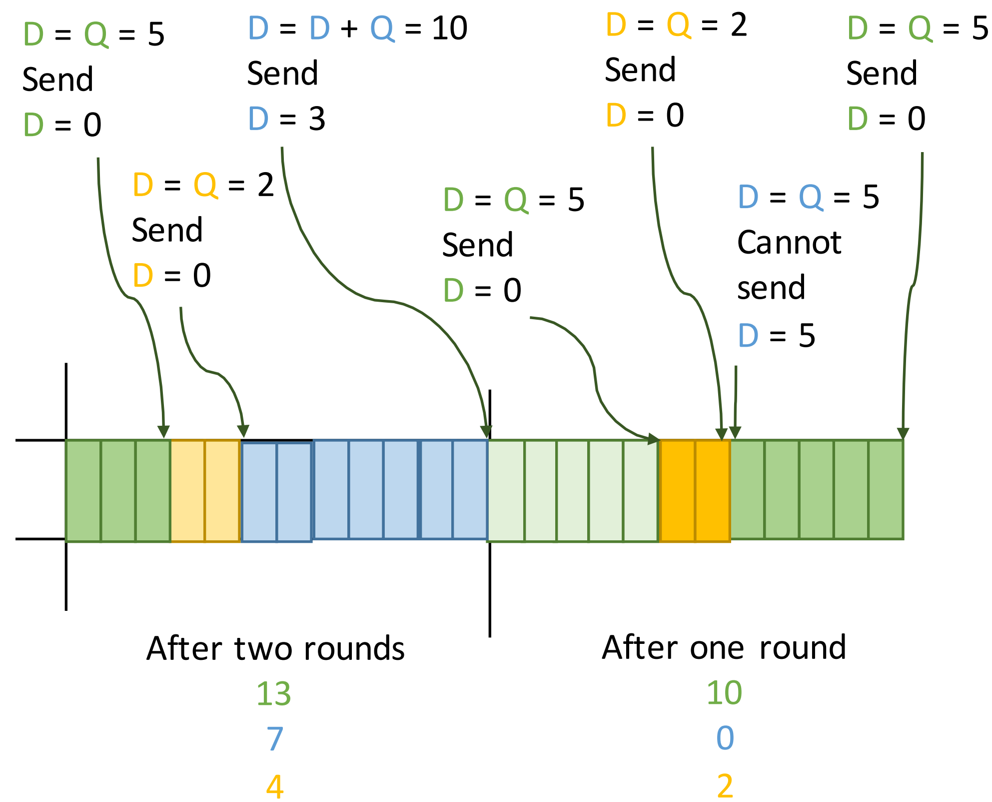
- That seems confusing… is there a simpler way?



- Define a quantum Q for each subqueue
- Define a counter C for each subqueue
  - How many bits allowed to be sent each turn

Deficit round robin





O(1), but less accurate

Overview

- What is scheduling?
  - FIFO
  - Round robin
  - Weighted round robin
- Generalized processor sharing (GPS)
- Implementations of GPS
  - Deficit round robin
  - Weighted fair queuing
- Rate control with GPS: Leaky bucket
- Paper discussion

Q: How should a common resource be shared between multiple users?

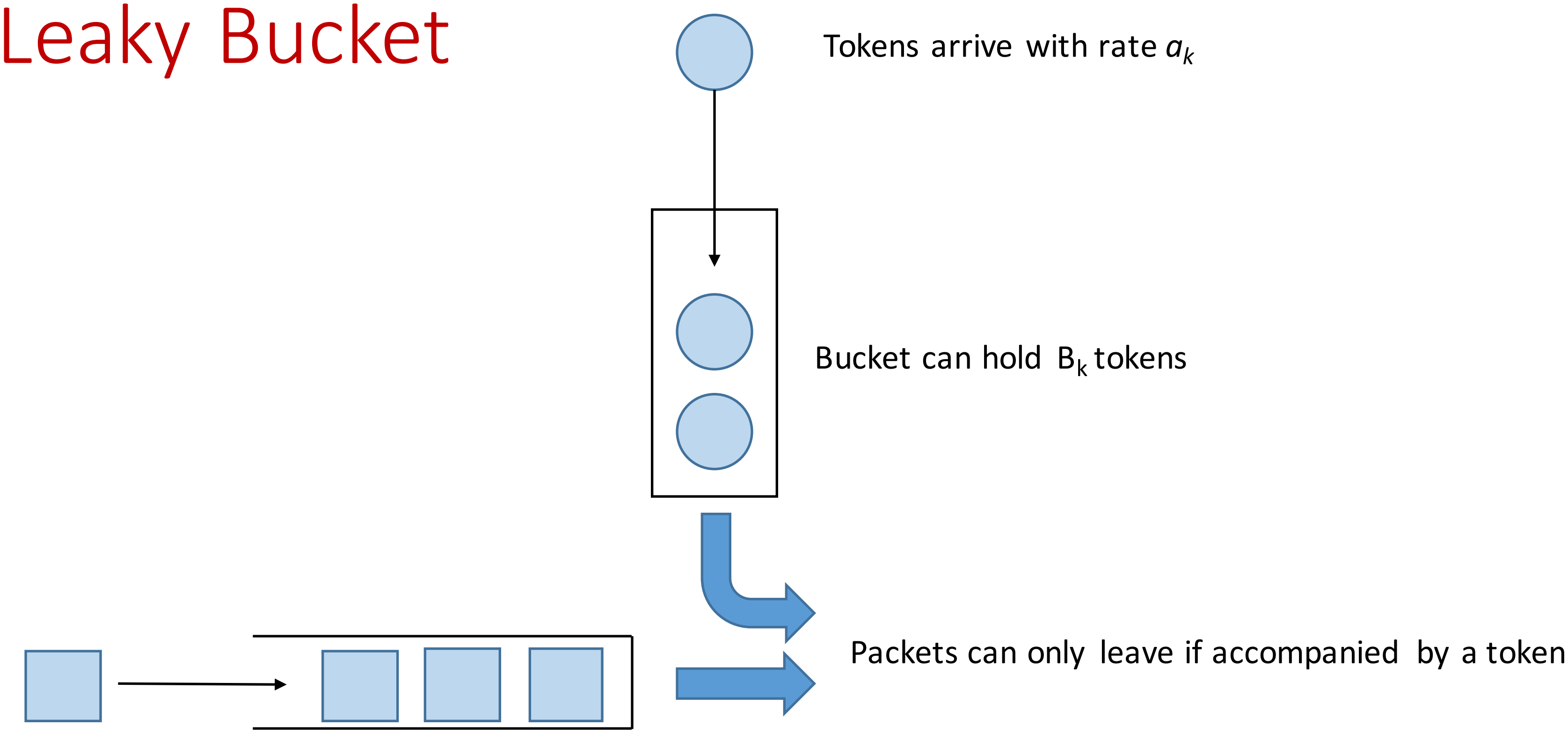
Performance guarantees with GPS

- GPS implementations give us an average throughput

$$\rho_k = \frac{w_k C}{\sum_j w_j}$$

- What about delay guarantee?

Leaky Bucket



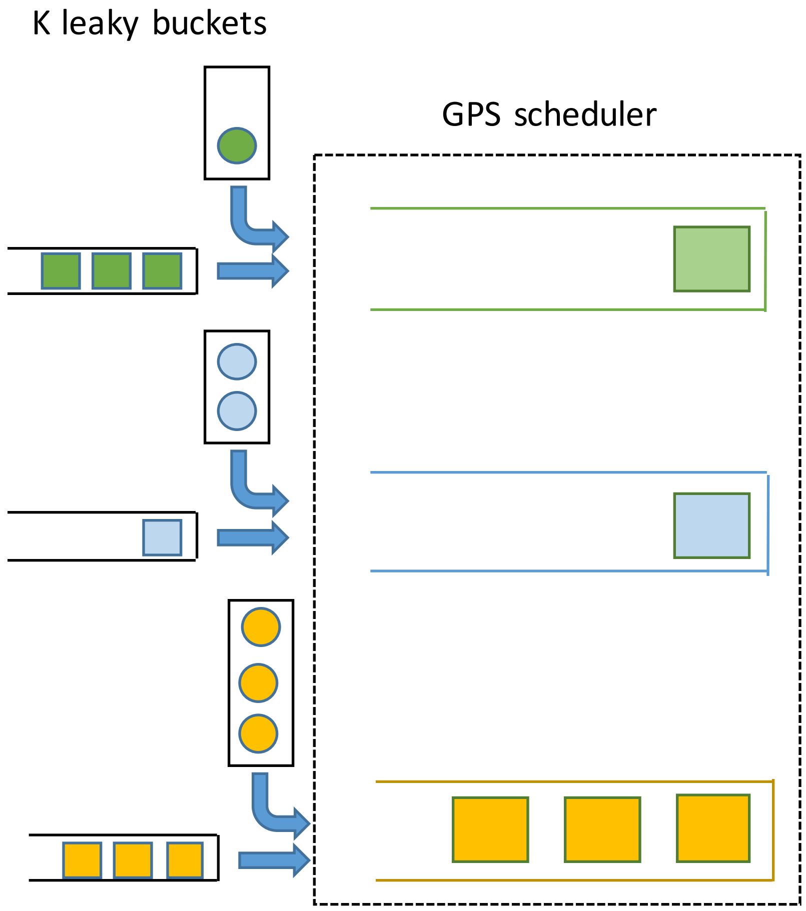
If the buffer is full, how many packets can leave during time interval T?

GPS + Leaky Bucket

- Max delay experienced by a bit:

$$\frac{B_k}{\rho_k}.$$

- Why?
  1. At most B_k packets waiting in GPS scheduler
  2. Delay = (B_k pkts) / (ρ_k pkts/s)



Sources

- *An Introduction to Computer Networks*, Peter Dordal
  http://intronetworks.cs.luc.edu/current/html/queuing.html

- A. Parekh and R. Gallager, "A Generalized Processor Sharing Approach to Flow Control in Integrated Services Networks: The Single-Node Case", *ToN* 1993.