

Wireless Networks

(CSC-7602)

Lecture 8

(15 Oct. 2007)

Seung-Jong Park (Jay)

<http://www.csc.lsu.edu/~sjpark>

1

Today

- Wireline Fair Scheduling
 - Why ?
 - Ideal algorithm
 - Practical algorithms
- Wireless Fair Scheduling
 - Why wireline algorithms cannot be used
 - Different approaches for wireless fair scheduling

2

Fair Scheduling

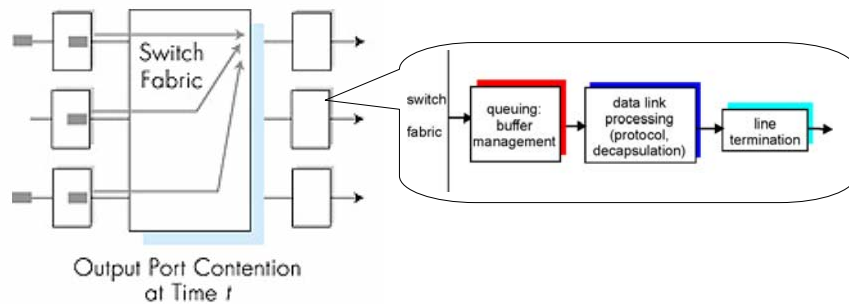
3

CSC7602 – S.J. Park

Queueing requires Scheduling

- **Queueing (Buffering) is required**

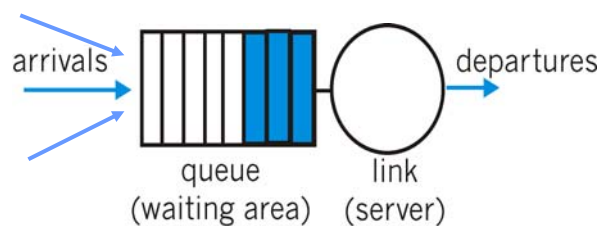
- when datagrams arrive from fabric faster than the transmission rate



4

Scheduling Mechanisms

- Scheduling happens at an output link of routers (switches)
 - Many flows transmitted simultaneously at an output link
 - Packet waiting for transmission are buffered
- Scheduling: choosing the next packet for transmission on a link can be done following a number of policies
- Simple case: FIFO
 - Single buffer
 - First packet will be transmitted when the link is ready to transmit



5

Scheduling Requirement

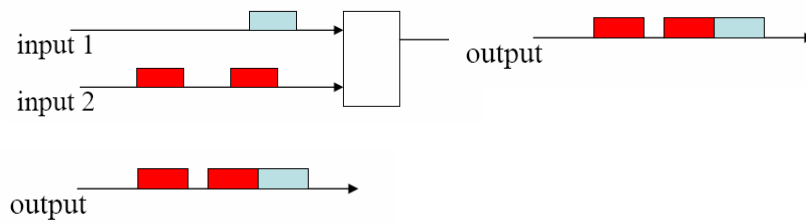
- Fair bandwidth allocation
 - for best effort traffic (TCP)
 - all competing flows receive the same amount of resource
 - E.g., email, some types of file transfer
 - Or some applications (real time and video on demand) require guaranteed service
 - E.g., packet voice, interactive video, stock quotes
- Easy implementation
 - Decision should be implementable in a few instructions

6

Classes of Scheduling

■ Work-conserving

- Server (output link) is never idle when there is packet waiting

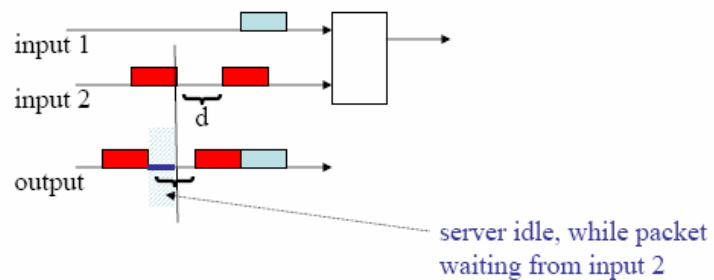


7

Classes of Scheduling (cont.)

■ Non-working conserving

- Server can be idle while packet is waiting
 - Little bandwidth lost
 - Why ? : to control burstiness and delay variation
- Finally, can change rate of input 2 to decouple scheduling and rate control



8

Scheduling Disciplines

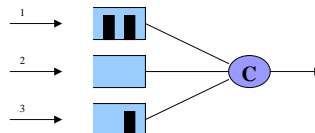
- Work-Conserving Scheduling Disciplines
 - Virtual Clock
 - Fair Queueing
 - Fluid Fair Queueing
 - Weighted Fair Queueing (WFQ)
 - Packet Generalized Processor Sharing (PGPS)
 - Worst Case Fair Weighted Fair Queueing (WF²Q)
 - Delay Earliest-Due-Date
 - Self -Clocked Fair Queueing (SCFQ)
 - Rotating Priority Queue (RPQ)
- Non-Work-Conserving Scheduling Disciplines
 - Jitter Earliest-Due-Date (Jitter EDD)
 - Stop & Go Queueing
 - Hierarchical Round Robin
 - Burst Scheduling

9

Fair Queueing

■ Definition

- If each connection defines a separate FCFS queue and k of these queues are currently not empty, then each non-empty queue receives $1/k$ -th of the link (server) bandwidth.



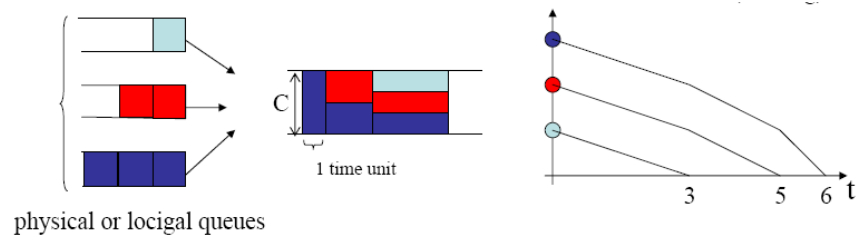
■ History

- Originally, fair queueing was adapted from a processor sharing for computer networks

10

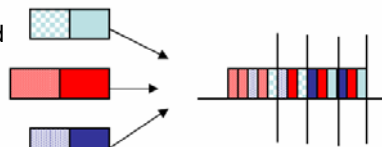
Fluid Fair Queueing

- Fluid approximation = *Generalized processor sharing (GPS)*
 - idealized policy bandwidth
 - assumption: dedicated flow, flows (like fluid)
 - used for not implementation but performance comparison
- Problem ?
 -
 -

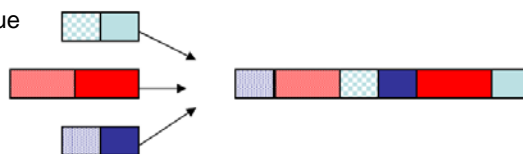


Practical Solution for Fair Queueing

- Bit by Bit Fair sharing
 - One bit from each backlogged queue at each round
 - round robin
 - Router can send not a bit but a packet

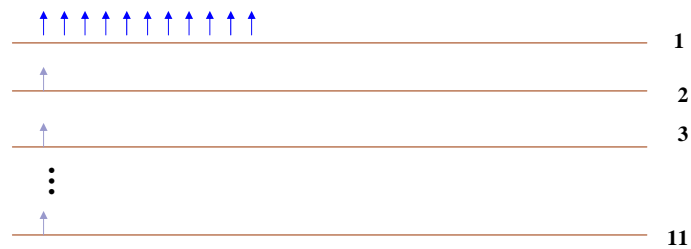


- Packet-level fair queueing
 - One packet from each queue
 - Round robin
 - Practical



Approximated Fair Queueing

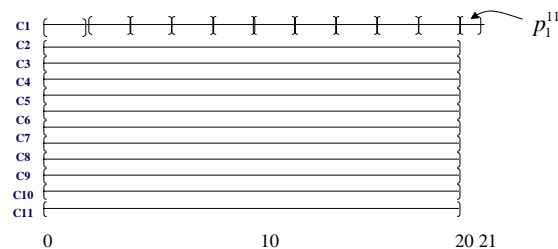
- Weighted Fair Queueing (WFQ) = Packetized GPS (PGPS)
 - The first cell that would complete service in the corresponding FFQ system is served first.
- Example
 - 11 connections of which link speed is 1
 - All packets have the same size of 1
 - Guaranteed rate for Connection 1 is 0.5, Connection 2-11 has 0.05
 - Connection 1 creates 11 back to back cells while 2 - 11 connections create only 1 cell at time 0.



13

Fluid Fair Queueing Order

- FFQ will take 2 time units to service each of the first 10 packets on connection 1, 1 time unit to service the 11-th packet from connection 1 and 20 time units to service the first packet from other connections.

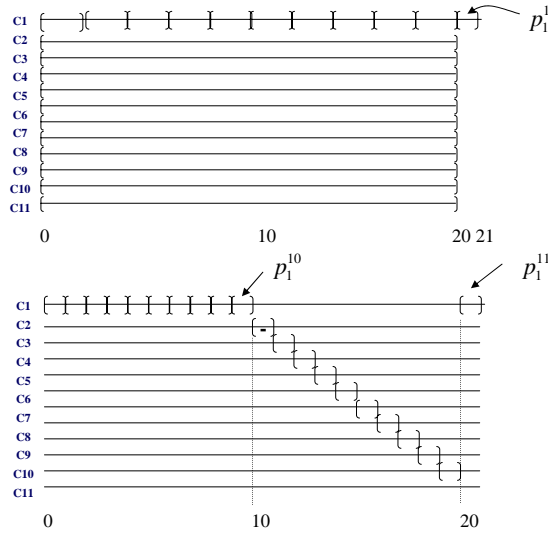


14

Weighted Fair Queueing

- If we denote the k-th packet on connection j as p_j^k , then the start and finish times of cells in FFQ system are:

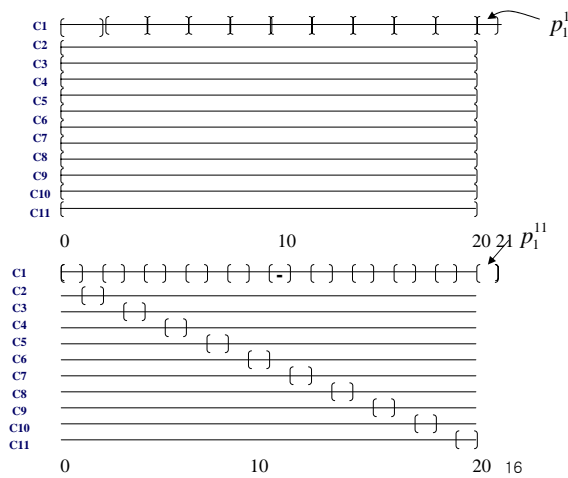
Start Time	Finish Time	Cell
$2(k-1)$	$2k$	p_1^k for $k = 1, \dots, 10$
20	21	p_1^{11}
0	20	p_j^i for $j = 2, \dots, 11$



Worst Case Fair Weighted Fair Queueing (WF²Q)

- While WFQ uses only finish times of cells in the corresponding FFQ system, WF²Q uses both START TIMES and FINISH TIMES of cells in the corresponding FFQ system.
- Example

- in case of tie among c2-c11, lower connection number will win



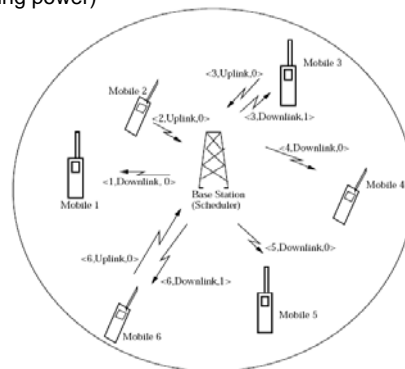
Fair Scheduling for Wireless Networks

17

CSC7602 – S.J. Park

Wireless Fair Scheduling

- Channel Characteristics of Wireless Cellular Network
 - Dynamically varying capacity
 - Location dependent channel errors and bursty errors
 - Scarce resources (battery & processing power)
 - Contention
 - No global state



18

Wireless Fair Scheduling

- Why do we need ?
 - Wireless channel capacities are scarce
 - Fair sharing of bandwidth becomes critical
 - Both short-term and long-term fairness important

- Do we need same wireline schedulings or new one ?
 - Location dependent and bursty errors
 - For the same wireless channel, a mobile station might experience a clean channel while another might experience high error rates.
 - Why?
 - Different distance and different mobility
 - In wireline fair queuing, the channel is either usable by all flows or unusable by any of the flows ...
 - But, in case of wireless, some flows cannot use channel due to wireless channel condition although they have packets to transmit

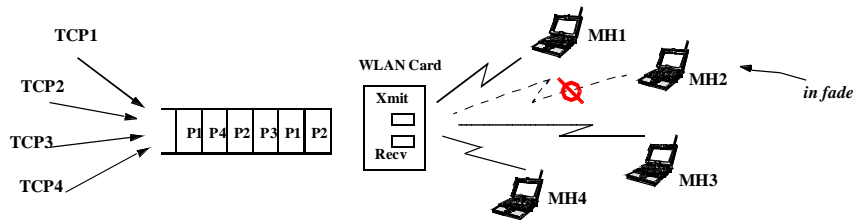
19

Terminology

- Service Model
 - Short term fairness
 - Long term fairness
 - Short term throughput bounds
 - Long term throughput bounds
 - Delay bounds for packets
- Terminology
 - Error free service
 - Leading flows
 - Lagging flows
 - In sync flows

20

Problems with FIFO Scheduling in MAC [Bhagwat96]

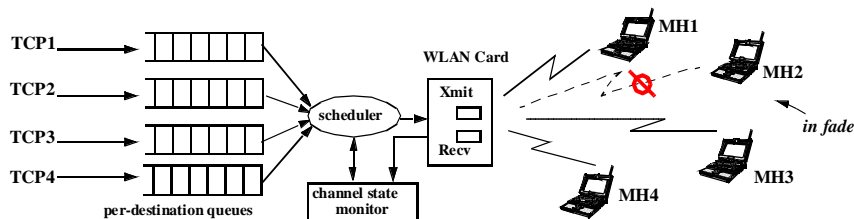


- **Burst errors may be spatially selective**
 - e.g. link to only one receiver may be under interference or in fade
- **During burst, all retransmission attempts to specific MH will fail**
 - burst errors observed to be 50-100 ms long in WLANs
- **FIFO is basically causing head of line blocking!**
 - other MHs starve even though link to them may be good
 - TCP to all MHs will increase RTT estimates, further increasing timeouts
 - poor resource utilization
 - fairness problem: MHs with bad link claiming more link time & b/w
 - a "fair" MAC is not enough in the presence of errors on the link

21

Channel State Dependent Scheduling

- **Primary culprits:**
 - CSMA/CA MAC makes repeated attempts even when channel is bad
 - FIFO dispatcher continues to send packets without regard to channel state
- **Solution: defer scheduled transmissions until next good period**
 - transmit packets for other destinations (those marked good) meanwhile burst periods for different MHs are independent
 - potential risk: TCP sender may timeout
 - but TCP timers \gg average burst durations
 - bad periods detected by radio feedback or multiple MAC transmit attempts
 - channels remain marked bad for an estimated burst interval length
 - round-robin scheduler (two sets: good & bad) worked best



22

Packet Fair Queuing in Wireless Links: The Problem

- **Traditional WFQ algorithms assume error free channel**
 - or, at least that either all flows can be scheduled or none
 - don't address fairness when a subset of backlogged flows cannot transmit because of bad channel
- **In wireless domain, a packet flow may experience location-dependent channel error and hence may not be able to transmit during a a given time window**
 - giving the channel to such flows is a waste and also not fair in terms of actual bandwidth they receive

Why WFQ Fails in Wireless Channels?

- **Example: backlogged flows during interval [0,2] with $r_1 = r_2 = r_3 = 1/3$**
 - flow 1 and flow 2 have error free channels
 - flow 3 has a channel error in the time interval [0,1]
 - applying FFQ equation over time periods [0,1] and [1,2] one gets

$$W_1[0, 1] = W_2[0, 1] = 1/2, \quad W_1[1, 2] = W_2[1, 2] = W_3[1, 2] = 1/3$$

- over [0,2] the allocation is $W_1[0, 2] = W_2[0, 2] = 5/6, \quad W_3[0, 2] = 1/3$

Not fair! (violates $\forall i, j \in B(t_1, t_2), \left| \frac{W_i(t_1, t_2)}{r_i} - \frac{W_j(t_1, t_2)}{r_j} \right| = 0$)

$B(t_1, t_2)$ is the set of flows that were backlogged in $[t_1, t_2]$

Solution: Wireless Fair Queuing

- **Goal**
 - Make short bursts of location-dependent channel errors transparent to users
- **Approach: dynamic reassignment of channel allocation over small time scales**
 - a backlogged flow f that perceives channel error during a time window $[t_1, t_2]$ is **compensated** over a later time window $[t_1', t_2']$ when f perceives a clean channel
- **Compensation mechanism: swap channel access & reclaim later**
 - grant additional channel access to f during $[t_1', t_2']$ in order to make up for the lost channel access during $[t_1, t_2]$
 - this additional channel access is granted to f at the expense of flows that were granted additional channel access during $[t_1, t_2]$ while f was unable to transmit any data
- **Many different proposals with different swapping mechanisms and flows between which swapping takes place, and different compensation models**

25

Problem in Defining Fairness

- Wireless channel: flow may be backlogged but unable to transmit due to channel error
 - should the flow be compensated at a later time?
 - should channel error be treated same as or differently from empty queue?
 - currently... either all flows are permitted to transmit or none
- Consider the scenario when flow f_1 and f_2 are both backlogged, but f_1 perceives a channel error while f_2 perceives a good channel
 - f_2 will additionally receive the share of the channel that would have been granted to f_1 in the error free case
- Question: should the fairness model readjust the service granted to f_1 and f_2 in a future time window in order to compensate f_1 ?
 - traditional fluid fair queuing model does not need to address this issue in a wireline model either all flows are permitted to transmit, or none

26

So, What is a Reasonable Model for Wireless Fair Service?

- Two fairness
 - Short-term fairness among flows that perceive a clean channel
 - Long-term fairness for flows with bounded channel error.
- Delay bounds for packets.
- Two throughput
 - Short-term throughput bounds for flows with clean channels
 - Long-term throughput bounds for all flows with bounded channel error.
- Support for both delay sensitive and error sensitive data flows.

27

Leading and Lagging Flow

- **Error-free service of a flow**
 - service that it would have received at the same time instant if all channels had been error-free, under identical offered load.
- **Leading flow**
 - if it has received channel allocation in excess of its error-free service.
- **Lagging flow**
 - if it has received channel allocation less than its error-free service.
- **Flow in sync**
 - neither leading nor lagging, it is said to be in sync
 - i.e. its channel allocation is exactly the same as its error-free service
- **How to compute lead and lag?**
 - wireless scheduling algorithm can explicitly simulate the error-free service and calculates the difference of the queue size of a flow in the error-free service and the queue size of the flow in reality

28

Adapting WFQ to Wireless

- Reference
 - Use an “error-free” fair service model with no channel error as reference
- Separation
 - Monitor and estimate channel condition for backlogged flows, and exclude those flows that have bad channels from consideration
- Calculate “lead” and “lag” for flows relative to the reference model
- Compensation for long term fairness
 - Compensate lagging flows that perceive a good channel at the expense of leading flows
 - Variations
 - No explicit compensation
 - Flow with maximum lag is given preference
 - Leading and lagging flows swap slots
 - Bandwidth is reserved for compensation

29

Many Algorithms...

- **Idealized Wireless Fair Queuing algorithm (IWFQ) [Lu97]**
- **IWFQ-variant: Wireless Packet Scheduling (WPS) [Lu97]**
- **CSDPS + Enhanced Class Based Queuing [Fragouli98]**
- **Channel-condition Independent Fair Queuing algorithm (CIF-Q) [Ng98]**
- **Server Based Fairness Approach (SBFA) [Ramanatha98]**
- **Wireless Fair Service algorithm (WFS) [Lu98]**

30

CSDPS [Fragouli98]

- CSDPS allows for the use of any error-free scheduling discipline – e.g. WRR with WFQ spread
- When a flow is allocated a slot and is not able to use it, CSDPS skips that flow and serves the next flow
- No measurement of lag or lead
- No explicit compensation model

- Lagging flows can thus make up lags only when leading flows cease to become backlogged or experience lossy channels sometime
- No long-term or short-term fairness guarantees

31

IWFQ [Lu97]

- WFQ is used for the error free service
- Packets tagged as in WFQ. Of the flows observing a clean channel, the flow with the minimum service tag packet is served
- Tags implicitly capture the service differences between flows (lagging flows will have a smaller service and hence will be scheduled earlier)

- Channel capture by lagging flows possible resulting in short term unfairness and starvation
- Even in-sync flows can become lagging during such capture periods
- Coarse short-term fairness guarantees because of possible starvation
- Provides long-term fairness

32

CIFQ

- STFQ (Start time fair queuing) used for the error free service
- Lag or lead computed as the difference between the actual service and the error free service
- A backlogged leading flow relinquishes slot with a probability p , a system parameter
- A relinquished slot is allocated to the lagging flow with the maximum normalized lag
- In-sync flows not affected since lagging flows use slots given up by leading flows
- Lagging flows can still starve leading flows under pathological scenarios
- Provides both short-term and long-term fairness

33

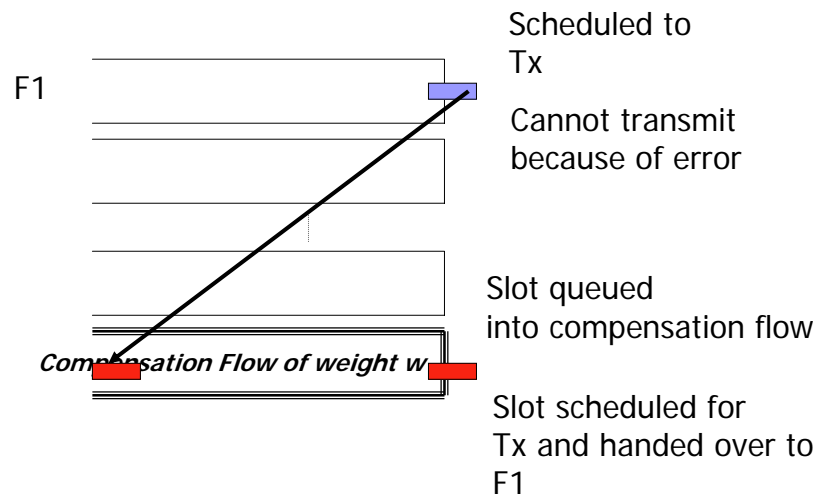
SBFA

- Any error free service model can be used
- SBFA reserves a fraction of the channel bandwidth statically for compensation by specifying a virtual compensation flow
- When a flow is unable to use a slot, it queues a slot-request to the compensation flow
- Scheduler serves compensation flow just as other flows
- When the compensation flow gets a slot, it turns the slot over to the flow represented by the head-of-line slot-request

- No concept of a leading flow
- All bounds supported by SBFA are only with respect to the remaining fraction of the channel bandwidth
- Performance of SBFA is sensitive to the statically reserved fraction
- No short-term fairness
- Long-term fairness dependent upon the reserved fraction

34

SBFA (Contd.)



35

Summary

- Wireline Fair Scheduling
 - Why ?
 - Ideal algorithm
 - Practical algorithms
- Wireless Fair Scheduling
 - Why wireline algorithms cannot be used
 - Different approaches for wireless fair scheduling
- Question ?

36