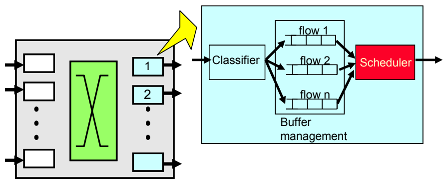
CEG 7450: Packet Scheduling

Packet Scheduling

- Decide when and what packet to send on output link
  - Usually implemented at output interface



Why Packet Scheduling?

- Can provide per flow or per aggregate protection
- Can provide absolute and relative differentiation in terms of
  - Delay
  - Bandwidth
  - Loss

Fair Queueing

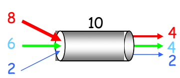
- In a fluid flow system it reduces to bit-by-bit round robin among flows
  - Each flow receives $min(r_i, f)$, where
    - r_i – flow arrival rate
    - f – link fair rate (see next slide)
- Weighted Fair Queueing (WFQ) – associate a weight with each flow [Demers, Keshav & Shenker '89]
  - In a fluid flow system it reduces to bit-by-bit round robin
- WFQ in a fluid flow system → Generalized Processor Sharing (GPS) [Parekh & Gallager '92]

4

Fair Rate Computation

- If link congested, compute f such that

$$\sum_i \min(r_i, f) = C$$



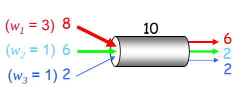
f = 4:
min(8, 4) = 4
min(6, 4) = 4
min(2, 4) = 2

5

Fair Rate Computation in GPS

- Associate a weight w_i with each flow i
- If link congested, compute f such that

$$\sum_i \min(r_i, f \times w_i) = C$$

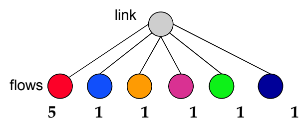


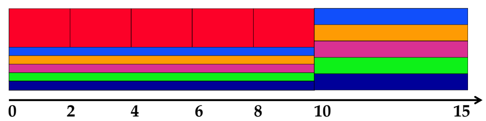
f = 2:
min(8, 2*3) = 6
min(6, 2*1) = 2
min(2, 2*1) = 2

6

Generalized Processor Sharing

- Red session has packets backlogged between time 0 and 10
- Other sessions have packets continuously backlogged





7

Generalized Processor Sharing

- A work conserving GPS is defined as

$$\frac{W_i(t,t+dt)}{w_i} = \frac{W(t,t+dt)}{\sum_{j \in B(t)} w_j} \qquad \forall i \in B(t)$$

- where
  - w_i – weight of flow i
  - $W_i(t_1, t_2)$ – total service received by flow i during $[t_1, t_2)$
  - $W(t_1, t_2)$ – total service allocated to all flows during $[t_1, t_2)$
  - $B(t)$ – set of backlogged flows

8

Generalized Processor Sharing

$$\frac{W_i(t_1,t_2)}{W_j(t_1,t_2)} \geq \frac{w_i}{w_j}, \qquad j = 1,2,\ldots,N$$

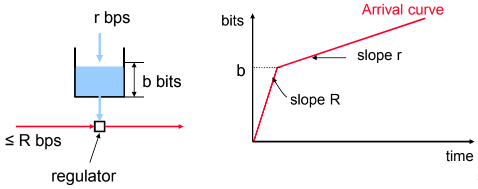
- where
  - w_i – weight of flow i
  - Flow i is continuously backlogged in $[t_1, t_2)$
  - $W_i(t_1, t_2)$ – total service received by flow i during $[t_1, t_2)$
- For any session i that is backlogged throughout the interval $[t_1, t_2]$

$$W_i(t_1,t_2) \geq \frac{(t_2 - t_1) r w_i}{\sum_j w_j}$$

9

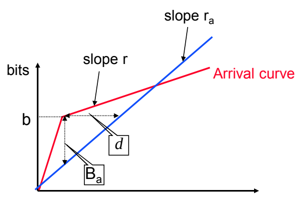
Flow Specification: Token Bucket

- Characterized by two parameters (*r*, *b*)
  - *r* – average rate
  - *b* – token depth
- Assume flow arrival rate ≤ *R* bps (e.g., *R* link capacity)
- A bit is transmitted only when there is an available token
- Arrival curve – maximum amount of bits transmitted by time *t*



Per-hop Reservation

- Given (*b*,*r*,*R*) and per-hop delay d
- Allocate bandwidth r_a and buffer space B_a such that to guarantee d



Properties of GPS

- End-to-end delay bounds for guaranteed service [Parekh and Gallager '93]
- Fair allocation of bandwidth for best effort service [Demers et al. '89, Parekh and Gallager '92]
- Work-conserving for high link utilization

Packet vs. Fluid System

- GPS is defined in an idealized fluid flow model
  - Multiple queues can be serviced simultaneously
- Real system are packet based systems
  - One queue is served at any given time
  - Packet transmission cannot be preempted
- Goal
  - Define packet algorithms approximating the fluid system
  - Maintain most of the important properties

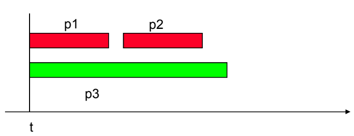
13

Packet Approximation of Fluid System

- Standard techniques of approximating fluid GPS
  - Select packet that finishes first in GPS assuming that there are no future arrivals
- Important properties of GPS
  - Finishing order of packets currently in system independent of future arrivals
- Implementation based on virtual time
  - Assign virtual finish time to each packet upon arrival
  - Packets served in increasing order of virtual finish times

14

Approximation of GPS - PGPS



- Packets that are delayed more in PGPS are those that arrive <u>too late</u> to be transmitted in their GPS order
- GPS departure order: p1, p2, p3
- PGPS departure order: p1, p3, p2

15

System Virtual Time

- Virtual time (V_{GPS}) – service that a backlogged flow with weight = 1 would receive in GPS

$$W_i(t,t+dt) = w_i \times \frac{W(t,t+dt)}{\sum_{j\in B(t)} w_j} \qquad \forall i \in B(t)$$

$$\frac{\partial W_i}{\partial t} = \frac{w_i}{\sum_{j\in B(t)} w_j} \times \frac{\partial W}{\partial t} \qquad \forall i \in B(t)$$

$$\frac{\partial V_{GPS}}{\partial t} = \frac{1}{\sum_{j\in B(t)} w_j} \times \frac{\partial W}{\partial t}$$

$$W_i(t_1,t_2) = w_i \times \int_{t=t_1}^{t_2} \left(\frac{1}{\sum_{j\in B(t)} w_j} \times \frac{\partial W}{\partial t} \right) dt \qquad \forall i \in B(t)$$

16

Service Allocation in GPS

- The service received by flow i during an interval $[t_1,t_2)$, while it is backlogged is

$$W_i(t_1,t_2) = w_i \times \int_{t=t_1}^{t_2} \frac{\partial V_{GPS}}{\partial t} dt \qquad \forall i \in B(t)$$

$$W_i(t_1,t_2) = w_i \times (V_{GPS}(t_2) - V_{GPS}(t_1)) \qquad \forall i \in B(t)$$

17

Virtual Time Implementation of Weighted Fair Queueing

$$V_{GPS}(0) = 0$$

$$S_j^k = F_j^{k-1} \quad \text{if session } j \text{ backlogged}$$

$$S_j^k = \max(F_j^{k-1}, V(a_j^k)) \quad \text{in general}$$

$$F_j^k = S_j^k + \frac{L_j^k}{w_j}$$

- a_j^k – *real* arrival time of packet k of flow j
- S_j^k – virtual starting time of packet k of flow j
- F_j^k – virtual finishing time of packet k of flow j
- L_j^k – length of packet k of flow j

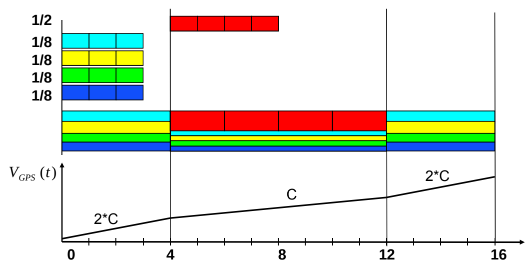
18

Virtual Time Implementation of Weighted Fair Queueing

- Need to keep per flow instead of per packet virtual start, finish time only
- System virtual time is used to reset a flow's virtual start time when a flow becomes backlogged again after being idle

19

System Virtual Time in GPS

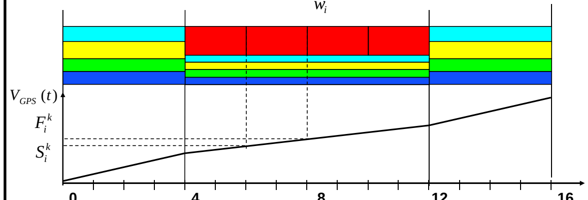


20

Virtual Start and Finish Times

- Utilize the time the packets would start S_i^k and finish F_i^k in a fluid system

$$F_i^k = S_i^k + \frac{L_i^k}{w_i}$$



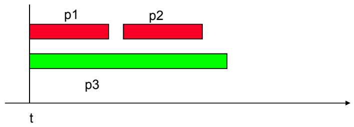
21

Approximation of GPS - PGPS

- GPS: server serves multiple session simultaneously and the traffic is fluid or infinitely divisible
- F_p: the time at which packet *p* will depart under GPS – virtual finish time
- PGPS
  - Serves packets in the increasing order of F_p
  - Serves the first packet that would complete service in the GPS simulation if no additional packet were to arrive afterwards

22

Approximation of GPS - PGPS



- Packets that are delayed more in PGPS are those that arrive <u>too late</u> to be transmitted in their GPS order
- GPS departure order: p1, p2, p3
- PGPS departure order: p1, p3, p2

23

Approximation of GPS - PGPS

- At time *t*, *p* and *p'* are packets in GPS system. Suppose *p* completes service before *p'* in GPS system if there are no arrivals after *t*, then *p* also completes service before *p'* for any pattern of arrivals after *t* in PGPS system
- L_{max}: the maximum packet length

$$F_p^{'} - F_p \le \frac{L_{\max}}{r}$$

$$W_i(0,t) - W_i'(0,t) \le L_{\max}$$

- The service received by a session *i* under PGPS could be far ahead of that received under GPS

24

Goals in Designing Packet based Fair Queueing Algorithms

- Improve worst-case fairness (see next):
  - Use Smallest Eligible virtual Finish time First (SEFF) policy
  - Examples: WF^2Q, WF^2Q+
- Reduce complexity
  - Use simpler virtual time functions
  - Examples: SCFQ, SFQ, DRR, FBFQ, leap-forward Virtual Clock, WF^2Q+
- Improve resource allocation flexibility
  - Service Curve

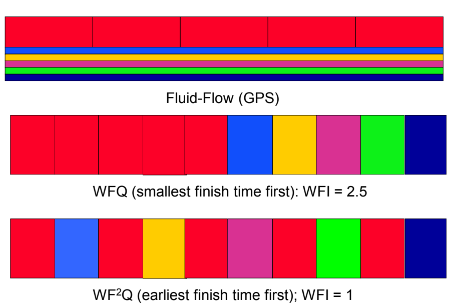
25

Worst-case Fair Index (WFI)

- Maximum discrepancy between the service received by a flow in the fluid flow system and in the packet system
- In WFQ, WFI = O(n), where n is total number of backlogged flows
- In WF^2Q, WFI = O(1)

26

WFI example



27

GPS

- Is work conserving
- Is a fluid model
- Service Guarantee
  - GPS discipline can provide an end-to-end bounded-delay service to a session whose traffic is constrained by a leaky bucket
- Fair Allocation
  - GPS can ensure fair allocation of bandwidth among all backlogged sessions regardless of whether or not their traffic is constrained. Thus, it is suitable for feedback based congestion control algorithms

28

PGPS

- Is a packet approximation algorithm of GPS
- Keeps the bounded-delay property of GPS

$$d^k_{i,PGPS} - d^k_{i,GPS} \le \frac{L_{\max}}{r} \quad \forall i,k \qquad (1)$$

$$W_{i,GPS}(0,\tau) - W_{i,PGPS}(0,\tau) \le L_{\max} \quad \forall i,\tau \qquad (2)$$

- Does not keep the worst-case fairness property of GPS, because the service received by a session *i* under PGPS could be far ahead of that received under GPS

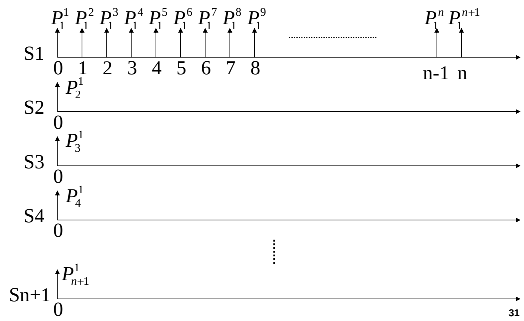
29

An Example

- Session 1 sends n+1 back-to-back packets starting at time 0, $\phi_1 = 0.5$.
- Session 2 to session n+1 send only one packet at time 0, $\phi_i = \frac{1}{2*n}, \; i = 2, \cdots, n+1$.
- All the packets size is $L_{\max}$, the transmission rate $r = L_{\max}$

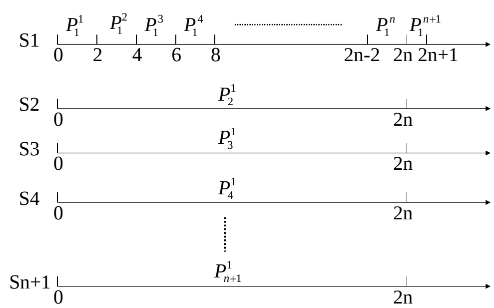
30

Packet Arrivals



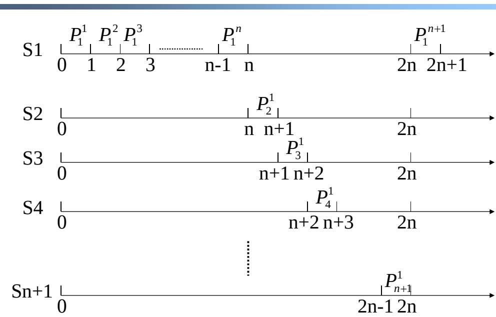
31

GPS Service Order



32

PGPS Service Order



33

Problem with PGPS

- The large discrepancy between GPS and PGPS is PGPS could provide service for a session far ahead of GPS.
- This large difference will result in unstable and less efficient network control algorithms.

34

Corresponding Systems

- Two queueing systems with different service disciplines are called corresponding systems of each other
  - if they have the same speed,
  - same set of sessions,
  - same arrival pattern, and
  - if applicable, same service share for each session.

35

Worst-case Packet Fair (I)

- To quantify the discrepancy between the services provided by a packet discipline and the GPS discipline
- A service discipline *s* is called worst-case fair for session *i* if for any time τ the delay of a packet arriving at τ is bounded above by $\frac{1}{r_i} Q_{i,s}(\tau) + C_{i,s}$, i.e.,

$$d_{i,s}^{k} < a_i^{k} + \frac{Q_{i,s}\left(a_i^{k}\right)}{r_i} + C_{i,s}$$

  where r_i is the throughput guarantee to session i, $Q_{i,s}(a_i^k)$ is the queue size of session *i* at time a_i^k and $C_{i,s}$ is a quantity independent of the queues of other sessions sharing the multiplexer

  A service discipline *s* is called worst-case fair if it is worst-case fair for all sessions.
- Normalized Worst-case Fair Index for session *i* at server *s*: $c_{i,s} = \frac{r_i C_{i,s}}{r}$
- Normalized Worst-case Fair Index for server *s* : $c_s = \max_i \{c_{i,s}\}$

36

Worst-case Packet Fair (II)

- GPS is worst-case fair with $c_{GPS} = 0$.
- c_{PGPS} may increase linearly as a function of number of session n

  In the example, consider the packet p_1^{n+1}. It won't depart until time $\frac{(2n+1)*L_{\max}}{r}$, thus

$$C_{1,PGPS} \ge d_{i,PGPS}^{n+1} - a_{i,PGPS}^{n+1} - \frac{Q\left(a_{i,PGPS}^{n+1}\right)}{r_i}$$

$$= (2n+1)\frac{L_{\max}}{r} - n\frac{L_{\max}}{r} - \frac{L_{\max}}{r/2}$$

$$= (n-1)\frac{L_{\max}}{r}$$

- The Normalized Worst-case Fair Index:

$$c_{PGPS} \ge C_{1,PGPS}\frac{r_1}{r} = \frac{n-1}{2}\frac{L_{\max}}{r}$$

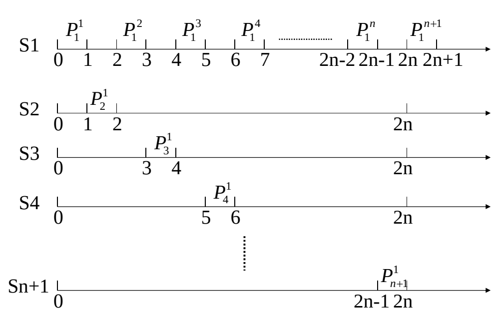
37

WF²Q

- Is a packet approximation algorithm of GPS
- WF²Q provides almost identical service with GPS, the maximum difference is no more than one packet size
- Difference between WF²Q and WFQ(PGPS)
  - In a WFQ system, when the server chooses the next packet for transmission at time *t*, it selects among all the packets that are backlogged at *t*, and selects the first packet that would complete service in the corresponding GPS
  - In a WF²Q system, when the server chooses the next packet at time *t*, it chooses only from the packets that have started receiving service in the corresponding GPS at *t*, and chooses the packet among them that would complete service first in the corresponding GPS

38

WF²Q Service Order



39

Properties of WF²Q

- Keeps the bounded-delay property of GPS
- Keeps the worst-case fair property of GPS

$$d^k_{i,WF2Q} - d^k_{i,GPS} \le \frac{L_{\max}}{r} \qquad (3)$$

$$W_{i,GPS}(0,\tau) - W_{i,WF2Q}(0,\tau) \le L_{\max} \qquad (4)$$

$$W_{i,WF2Q}(0,\tau) - W_{i,GPS}(0,\tau) \le \left(1 - \frac{r_i}{r}\right) L_{i,\max} \qquad (5)$$

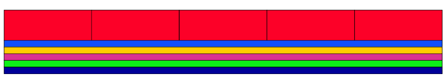
$$C_{i,WF2Q} = \frac{L_{i,\max}}{r_i} - \frac{L_{i,\max}}{r} + \frac{L_{\max}}{r} \qquad (6)$$

If all packets size is L, the Normalized

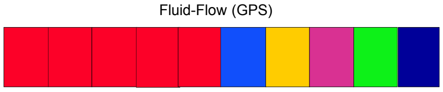
Worst-case Fair Index $c_{WF2Q} = \frac{L}{r}$.

40

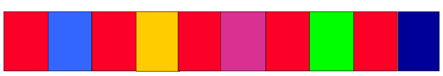
WFI example



Fluid-Flow (GPS)



WFQ (smallest finish time first): WFI = 2.5



WF²Q (earliest finish time first); WFI = 1

41

Rate-controlled Server

- Rate-controlled server = Regulators + Scheduler
- Packets are held in the regulators until their eligibility time before they are passed to the scheduler
  - Different policies of assigning eligibility times result in different regulators.
- The scheduler only schedules eligible packets
- R-WFQ and R-GPS have the same regulators but different schedulers

The eligibility time for the K^{th} packet on session i is:

$e_i^k = b_{i,GPS}^k$ (the service start time for the packet in GPS)

The schedulers for R-WFQ and R-GPS are WFQ(WFQ*) and GPS(GPS*) respectively.

42

Two Lemmas

- ***Lemma 1***
  An R-GPS system is equivalent to its corresponding GPS system. i.e., for any arrival sequence, the instantaneous service rates for each connection at any given time are exactly the same with either service discipline, and $d^k_{i,GPS} = d^k_{i,R-GPS}$ holds.
- ***Lemma 2***
  An R-WFQ system is equivalent to the corresponding WF^2Q system. i.e., for any arrival sequence, packets are serviced in exactly the same order with either service discipline and $d^k_{i,WF2Q} = d^k_{i,R-WFQ}$ holds.

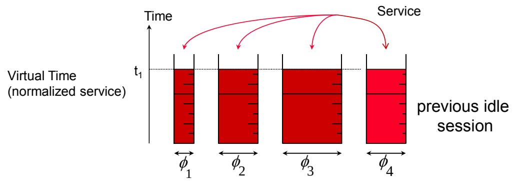
Problem with WF^2Q

- The time complexity of implementing WF^2Q is high
- Because it is based on a virtual time function which is defined with respect to the corresponding GPS system
- It leads to considerable computational complexity due to the need for simulating events in the GPS system

Summary

- WF^2Q is a better packet approximation algorithm of GPS than WFQ
- It provides almost identical service with GPS; the maximum difference is no more than one packet size
- The problem with WF^2Q is the time complexity for computing the virtual time

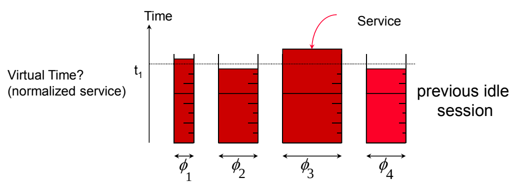
Virtual Time in Fluid GPS System



- Virtual time function represents the cumulative fair amount of service
- A previously idle session starts to receive service at the virtual time level when it becomes active
  - no compensation for the idle period
  - start receiving service immediately after becoming active
- Accurate but difficult to compute
  - need to emulate fluid system, worst case complexity O(N)

46

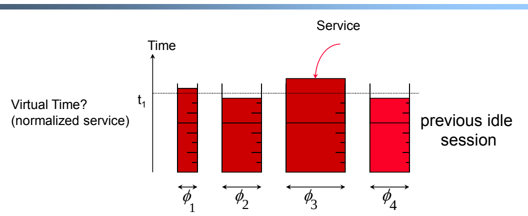
Virtual Time in Packet System



- Can we compute system virtual time based on information in the packet system?
  - no need to emulate fluid system
- Normalized amounts of service received by backlogged sessions are different
- Which level to set the previous idle session?

47

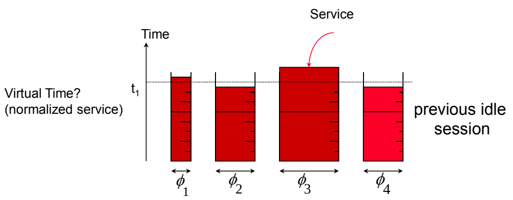
Virtual Time in Packet System



- Set too low: newly backlogged session receives more than its fair share, performance for other sessions degrade
- Set too high: performance for newly backlogged session degrade
- Tradeoff between accuracy and simplicity

48

Heuristics of Computing Virtual Time in Packet System



- SCFQ: virtual finish time of packet being serviced
- SFQ: virtual start time of packet being serviced
- WF^2Q+: $v(t+\tau)=\max\{v(t)+\tau, \min_{j\in B(t+\tau)} Sj\}$

49

WF^2Q+

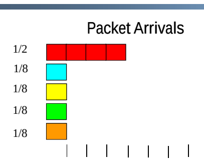
- WFQ and WF^2Q
  - Need to emulate fluid GPS system
  - High complexity
- WF^2Q+
  - Provide same delay bound and WFI as WF^2Q
  - Lower complexity
- Key difference: virtual time computation

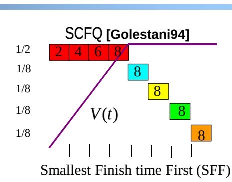
$$V_{WF^2Q+}(t+\tau)=\max(V_{WF^2Q+}(t)+W(t,t+\tau), \min_{i\in B(t+\tau)}(S_i^{h_i(t+\tau)}))$$

  - $h_i(t+\tau)$- sequence number of the packet at the head of the queue of flow i
  - $S_i^{h_i(t+\tau)}$ - virtual starting time of the packet at the head of queue i
  - $B(t)$ - set of packets backlogged at time t in the packet system

50

Self Clocked Fair Queuing (SCFQ)





- System virtual time function
  - the finish time of the current packet being serviced
- Packet selection policy
  - Smallest virtual Finish time First (SFF)

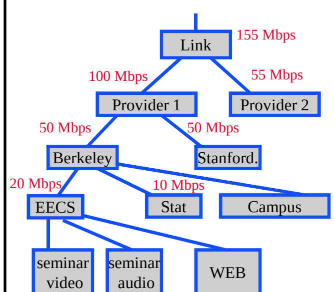
51

Start Time Fair Queuing (SFQ)

- System virtual time function
  - the start time of the current packet being serviced
- Packet selection policy
  - Smallest virtual Start time First (SSF)

52

Hierarchical Resource Sharing

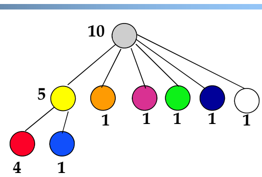


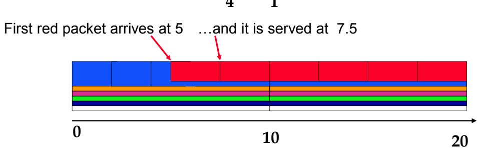
- Resource contention/sharing at different levels
- Resource management policies should be set at different levels, by different entities
  - Resource owner
  - Service providers
  - Organizations
  - Applications

53

Hierarchical-GPS Example

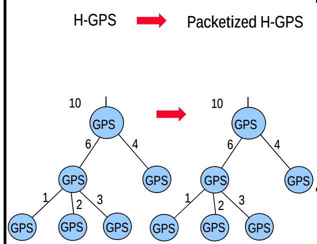
- Red session has packets backlogged at time 5
- Other sessions have packets continuously backlogged





54

Packet Approximation of H-GPS

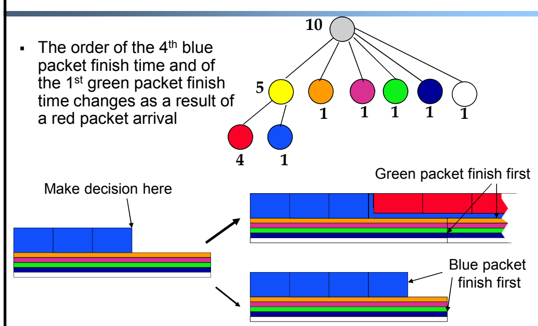


- Idea 1
  - Select packet finishing first in H-GPS assuming there are no future arrivals
  - Problem:
    - Finish order in system dependent on future arrivals
    - Virtual time implementation won't work
- Idea 2
  - Use a hierarchy of PFQ to approximate H-GPS

55

Problems with Idea 1

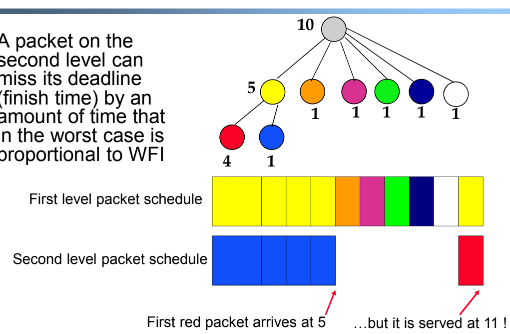
- The order of the 4th blue packet finish time and of the 1st green packet finish time changes as a result of a red packet arrival



56

Hierarchical-WFQ Example

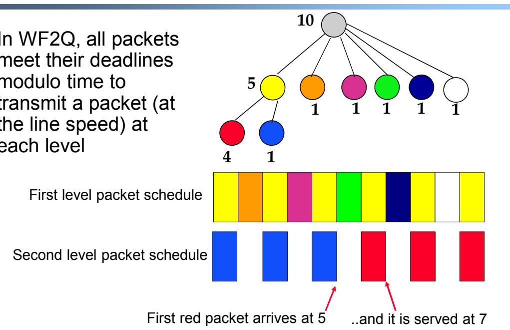
- A packet on the second level can miss its deadline (finish time) by an amount of time that in the worst case is proportional to WFI



57

Hierarchical-WF²Q Example

- In WF2Q, all packets meet their deadlines modulo time to transmit a packet (at the line speed) at each level



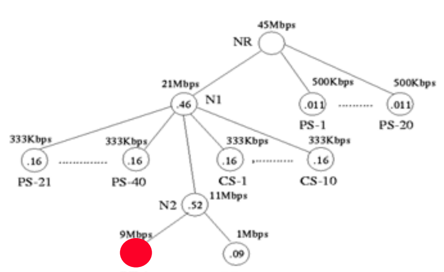
WF²Q+

- WFQ and WF²Q
  - Need to emulate fluid GPS system
  - High complexity
- WF²Q+
  - Provide same delay bound and WFI as WF²Q
  - Lower complexity
- Key difference: virtual time computation

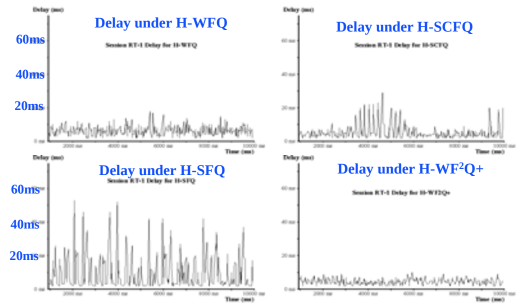
$$V_{WF^2Q+}(t+\tau) = \max(V_{WF^2Q+}(t) + W(t,t+\tau), \min_{i \in B(t+\tau)}(S_i^{h_i(t+\tau)}))$$

  - $h_i(t+\tau)$- sequence number of the packet at the head of the queue of flow i
  - $S_i^{h_i(t+\tau)}$ - virtual starting time of the packet at the head of queue i
  - $B(t)$ - set of packets backlogged at time t in the packet system

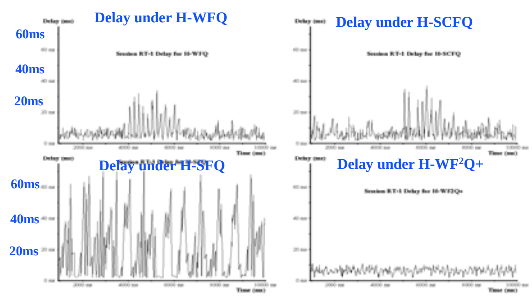
Example Hierarchy



Uncorrelated Cross Traffic



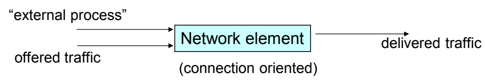
Correlated Cross Traffic



Why Service Curve?

- WFQ, WF^{2}Q, H-WF2Q+
  - Guarantee a minimum rate: $\geq C \times w_i / \sum_{j=1}^{N} w_j$
    - N – total number of flows
  - A packet is served no later than its finish time in GPS (H-GPS) modulo the sum of the maximum packet transmission time at each level
- For better resource utilization we need to specify more sophisticated services (example to follow shortly)
- Solution: QoS Service curve model

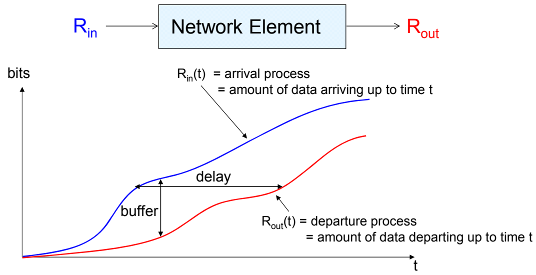
What is a Service Model?



- The QoS measures (delay,throughput, loss, cost) depend on offered traffic, and possibly other external processes.
- A service model attempts to characterize the relationship between offered traffic, delivered traffic, and possibly other external processes.

64

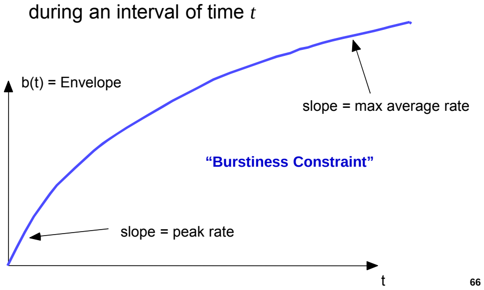
Arrival and Departure Process



65

Traffic Envelope (Arrival Curve)

- Maximum amount of service that a flow can send during an interval of time *t*



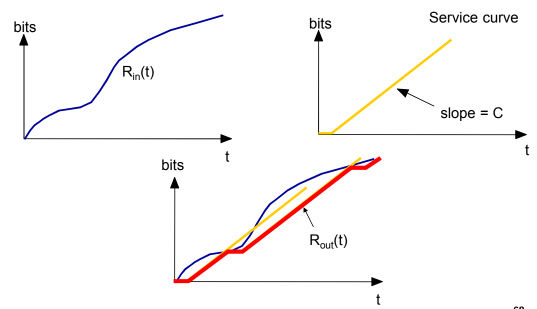
66

Service Curve

- Assume a flow that is idle at time s and it is backlogged during the interval (s, t)
- Service curve: the minimum service received by the flow during the interval (s, t)

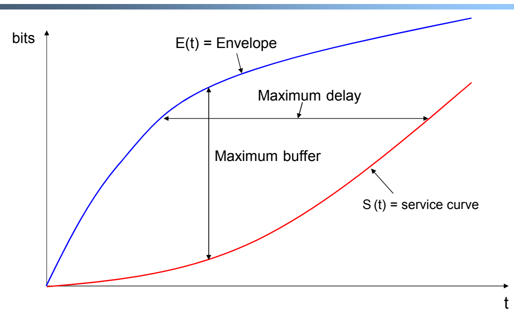
67

Big Picture



68

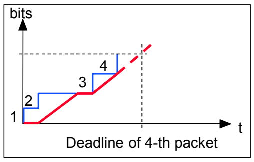
Delay and Buffer Bounds



69

Service Curve-based Earliest Deadline (SCED)

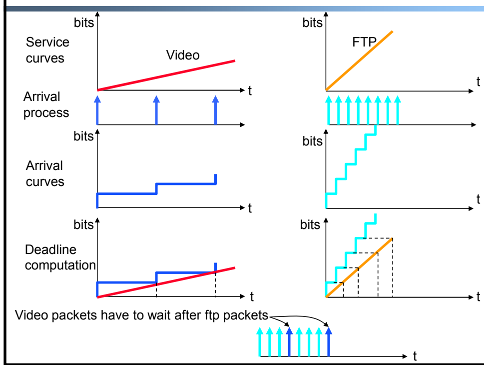
- Packet deadline – time at which the packet would be served assuming that the flow receives no more than its service curve
- Serve packets in the increasing order of their deadlines



- Properties
  - If sum of all service curves <= $C*t$
  - All packets will meet their deadlines modulo the transmission time of the packet of maximum length, i.e., L_{max}/C

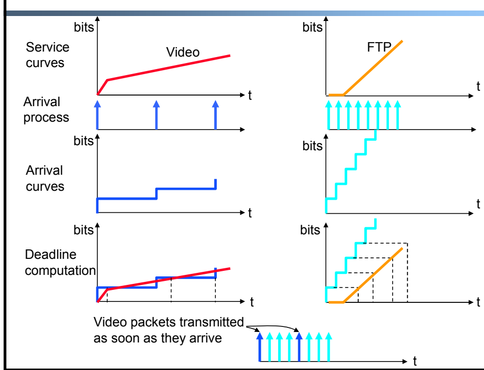
70

Linear Service Curves: Example



71

Non-Linear Service Curves: Example



72

Summary

- WF^2Q+ guarantees that each packet is served no later than its finish time in GPS modulo transmission time of maximum length packet
  - Support hierarchical link sharing
- SCED guarantees that each packet meets its deadline modulo transmission time of maximum length packet
  - Decouple bandwidth and delay allocations
- Question: does SCED support hierarchical link sharing?
  - No (why not?)
- Hierarchical Fair Service Curve (H-FSC) [Stoica, Zhang & Ng '97]
  - Support nonlinear service curves
  - Support hierarchical link sharing

73