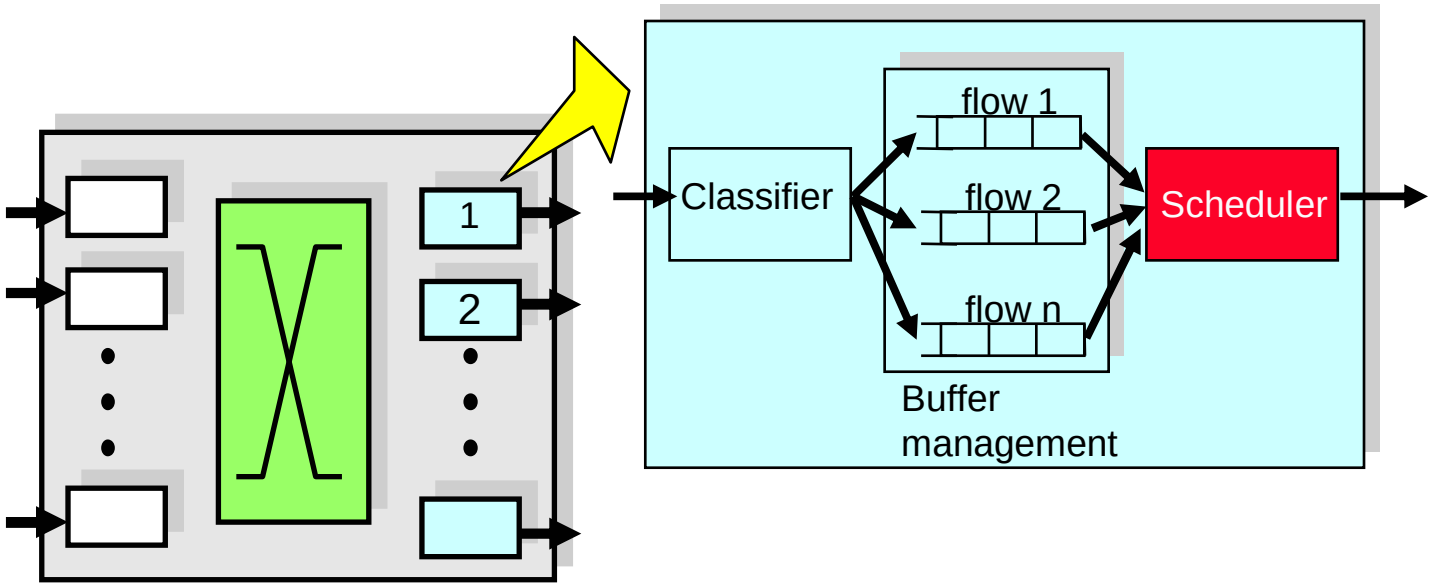
CS 268: Packet Scheduling

Ion Stoica

March 18/20, 2003

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

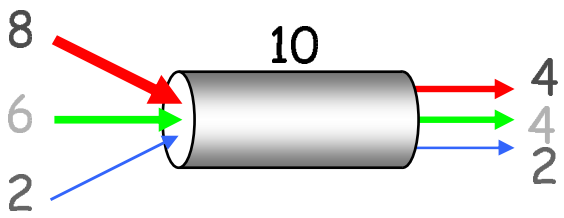
Fair Queueing

- 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]

Fair Rate Computation

- If link congested, compute f such that

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

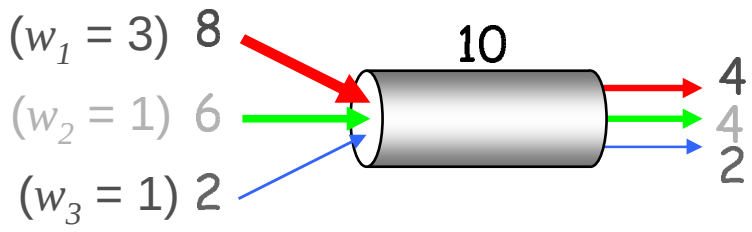


f = 4:
min(8, 4) = 4
min(6, 4) = 4
min(2, 4) = 2

Fair Rate Computation in GPS

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

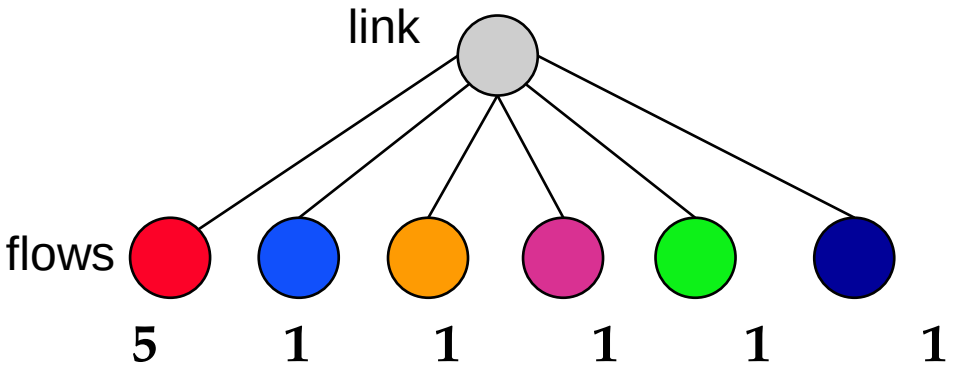
$$\sum_i \min(r_i, f \times w_i) = C$$

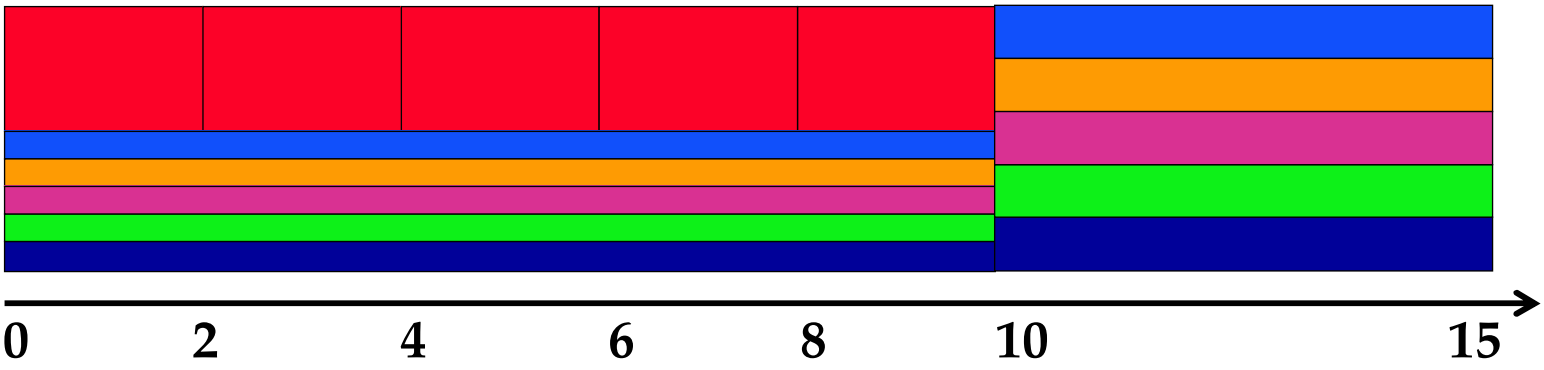


f = 2:
min(8, 2*3) = 6
min(6, 2*1) = 2
min(2, 2*1) = 2

Generalized Processor Sharing

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





Generalized Processor Sharing

- A work conserving GPS is defined as

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

- 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 al flows during $[t_1, t_2)$
  - $B(t)$ – number of backlogged flows

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 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

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 times

System Virtual Time

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

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

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

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

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

Service Allocation in GPS

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

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

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

Virtual Time Implementation of Weighted Fair Queueing

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

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

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

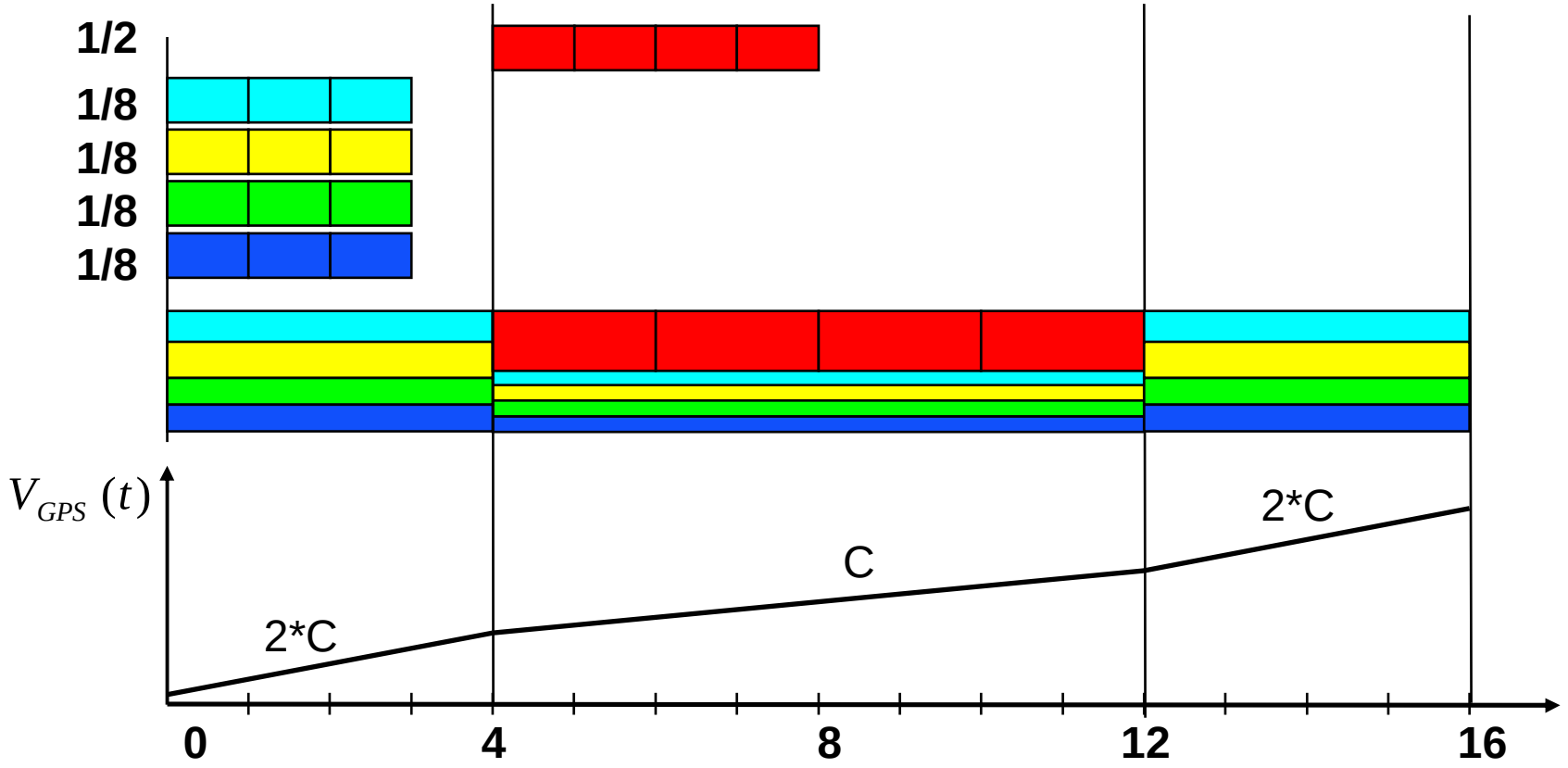
$$F_j^k = S_j^k + \frac{L_j^k}{w_j}$$

- a_j^k – 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

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

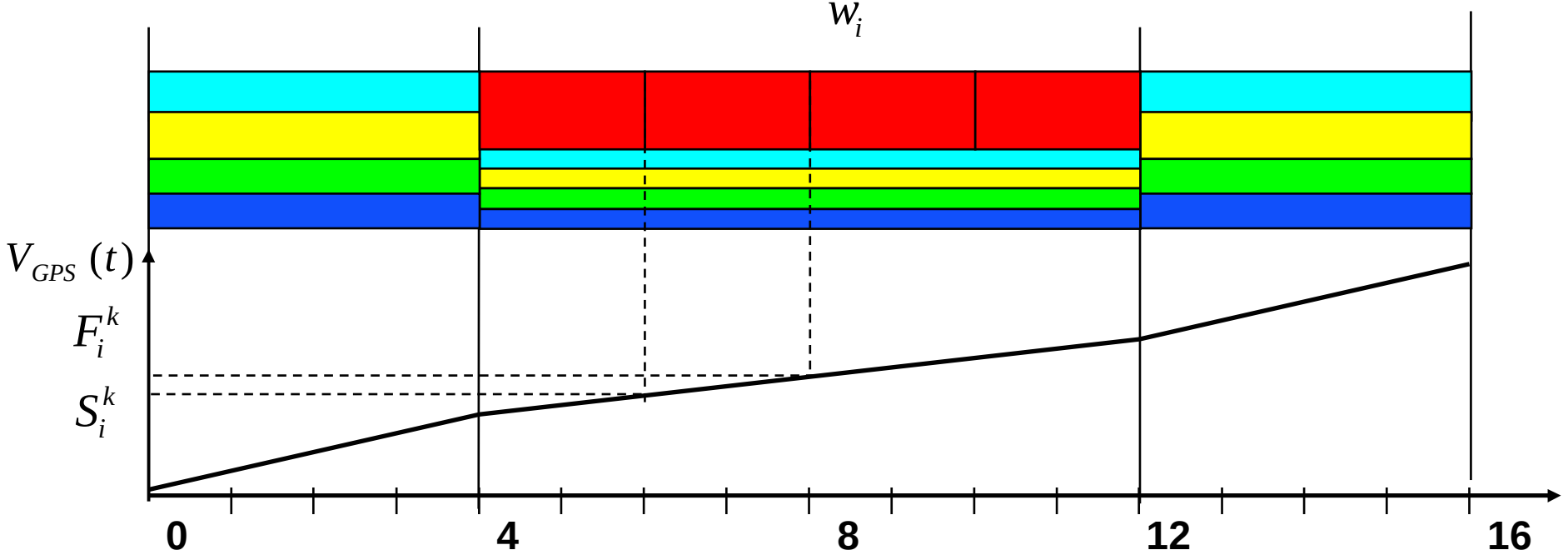
System Virtual Time in GPS



Virtual Start and Finish Times

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

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



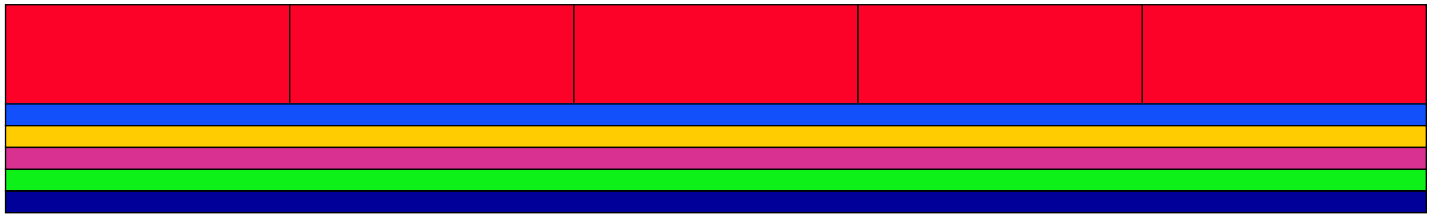
Goals in Designing Packet Fair Queueing Algorithms

- 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

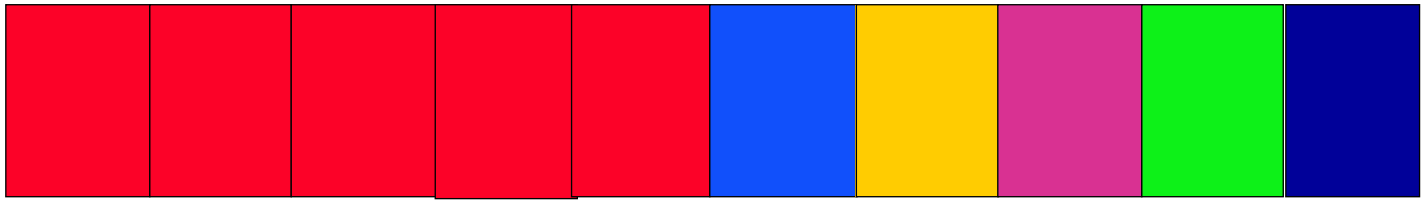
Worst-case Fair Index (WFI)

- 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 WF2Q, WFI = 1

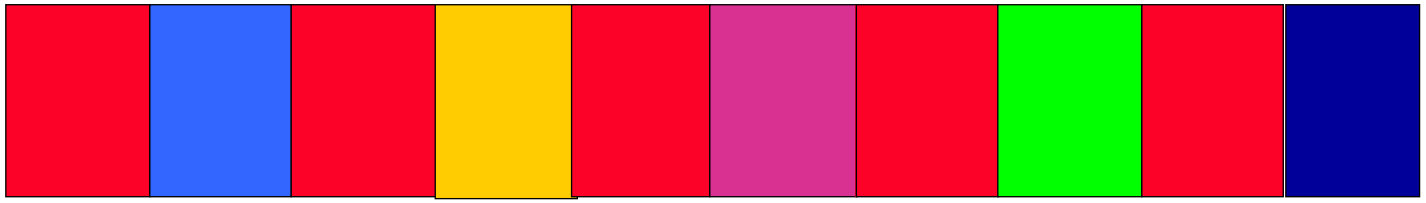
WFI example



Fluid-Flow (GPS)

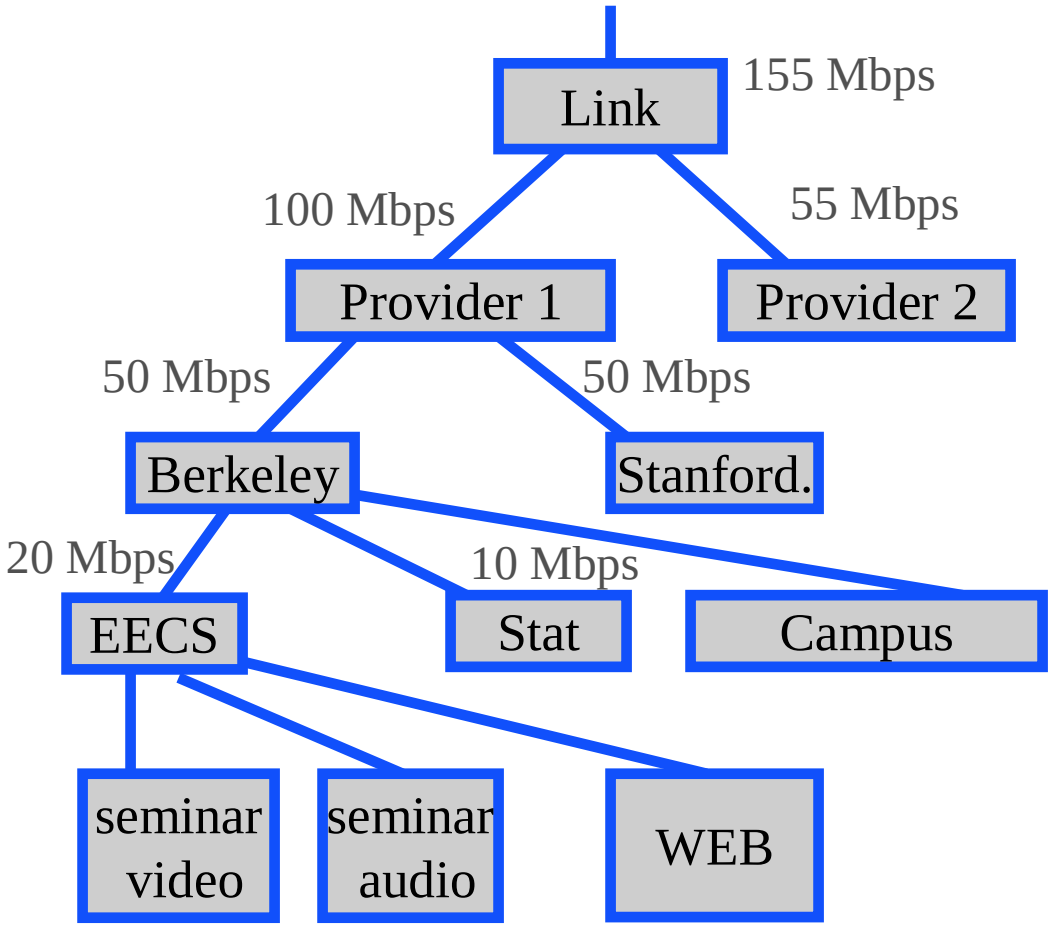


WFQ (smallest finish time first): WFI = 2.5



WF2Q (earliest finish time first); WFI = 1

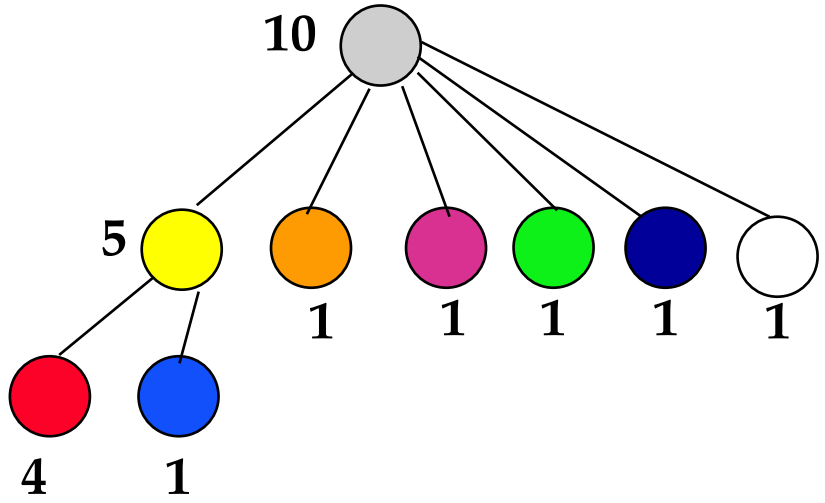
Hierarchical Resource Sharing



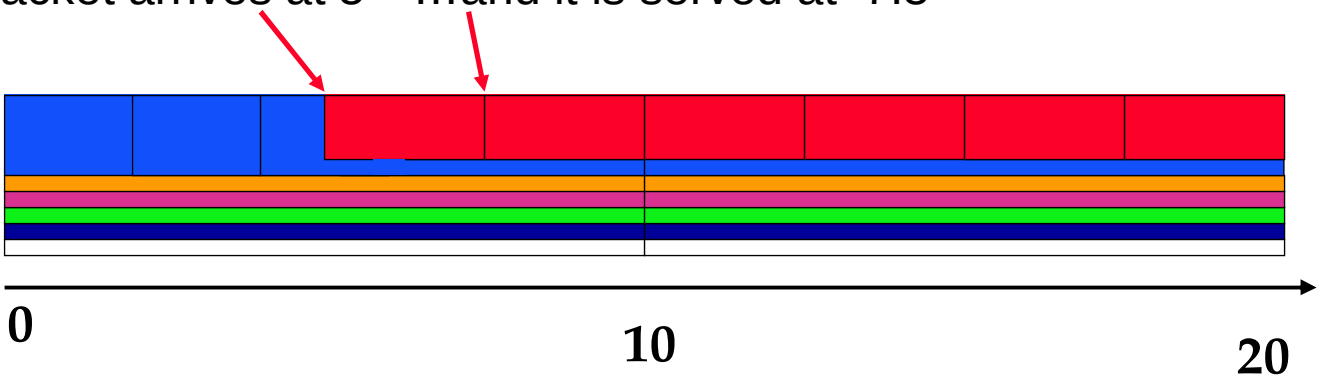
- Resource contention/sharing at different levels
- Resource management policies should be set at different levels, by different entities
  - Resource owner
  - Service providers
  - Organizations
  - Applications

Hierarchical-GPS Example

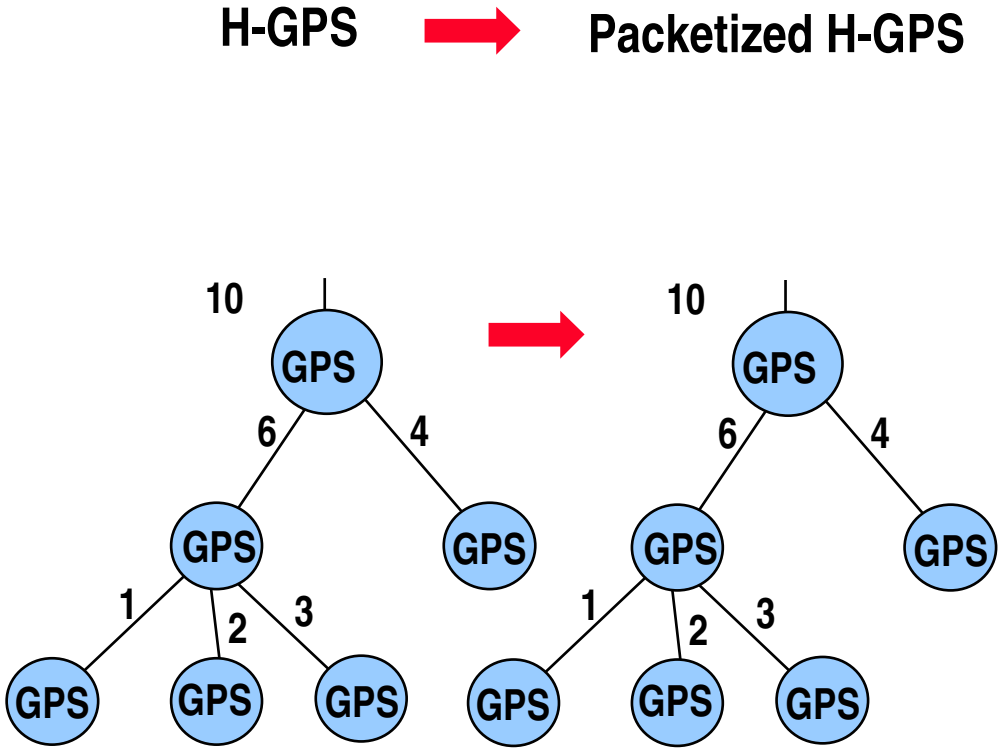
- Red session has packets backlogged at time 5
- Other sessions have packets continuously backlogged



First red packet arrives at 5 …and it is served at 7.5



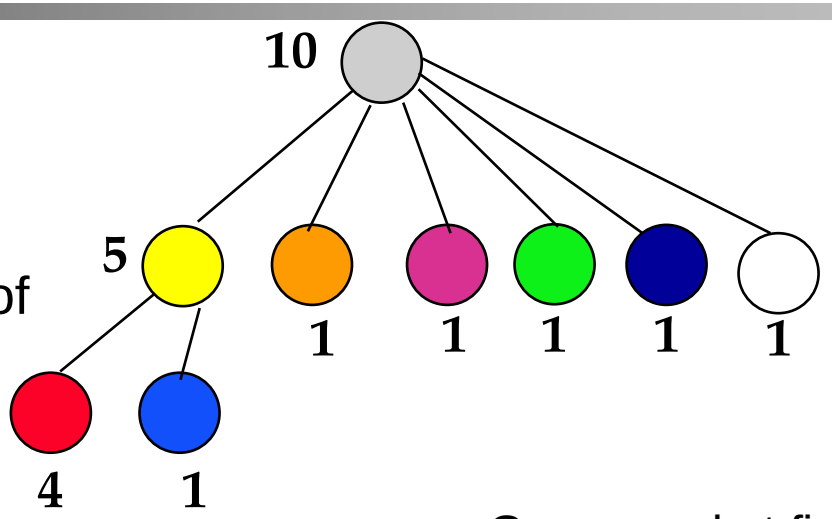
Packet Approximation of H-GPS

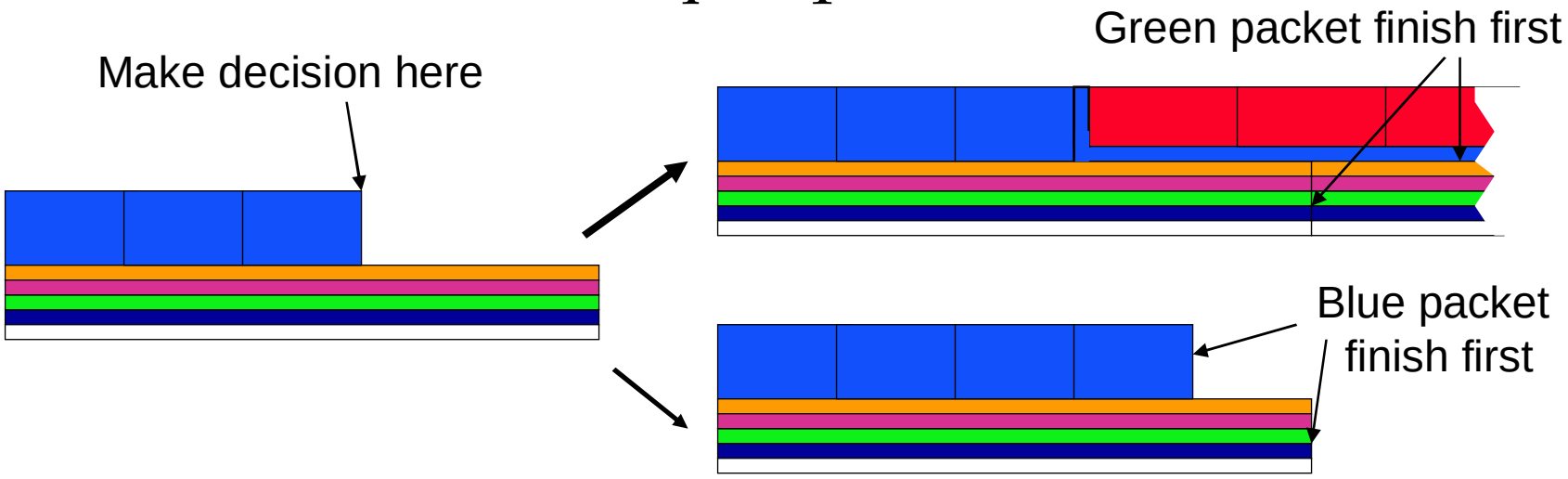


- Idea 1
  - Select packet finishing first in H-GPS assuming there are no future arrivals
  - Problem:
    - Finish order in system dependent on future arrivals
    - Virtual time implementation won't work
- Idea 2
  - Use a hierarchy of PFQ to approximate H-GPS

Problems with Idea 1

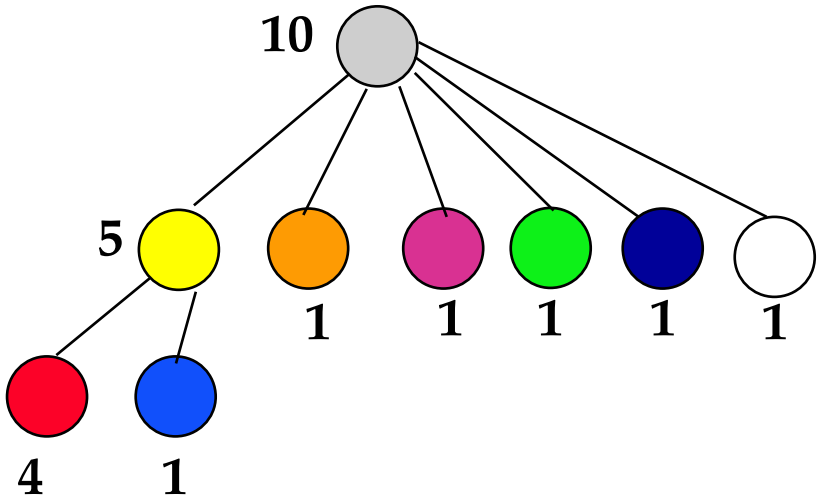
- The order of the 4th blue packet finish time and of the 1st green packet finish time changes as a result of a red packet arrival

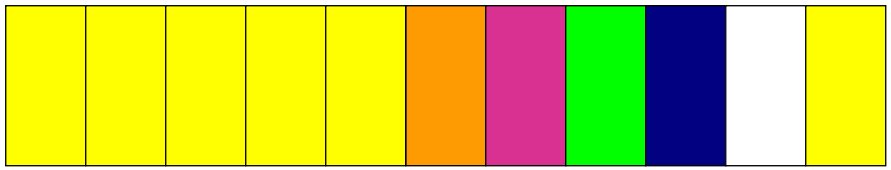




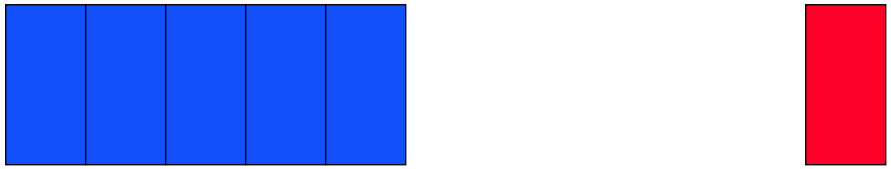
Hierarchical-WFQ Example

- 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





First level packet schedule



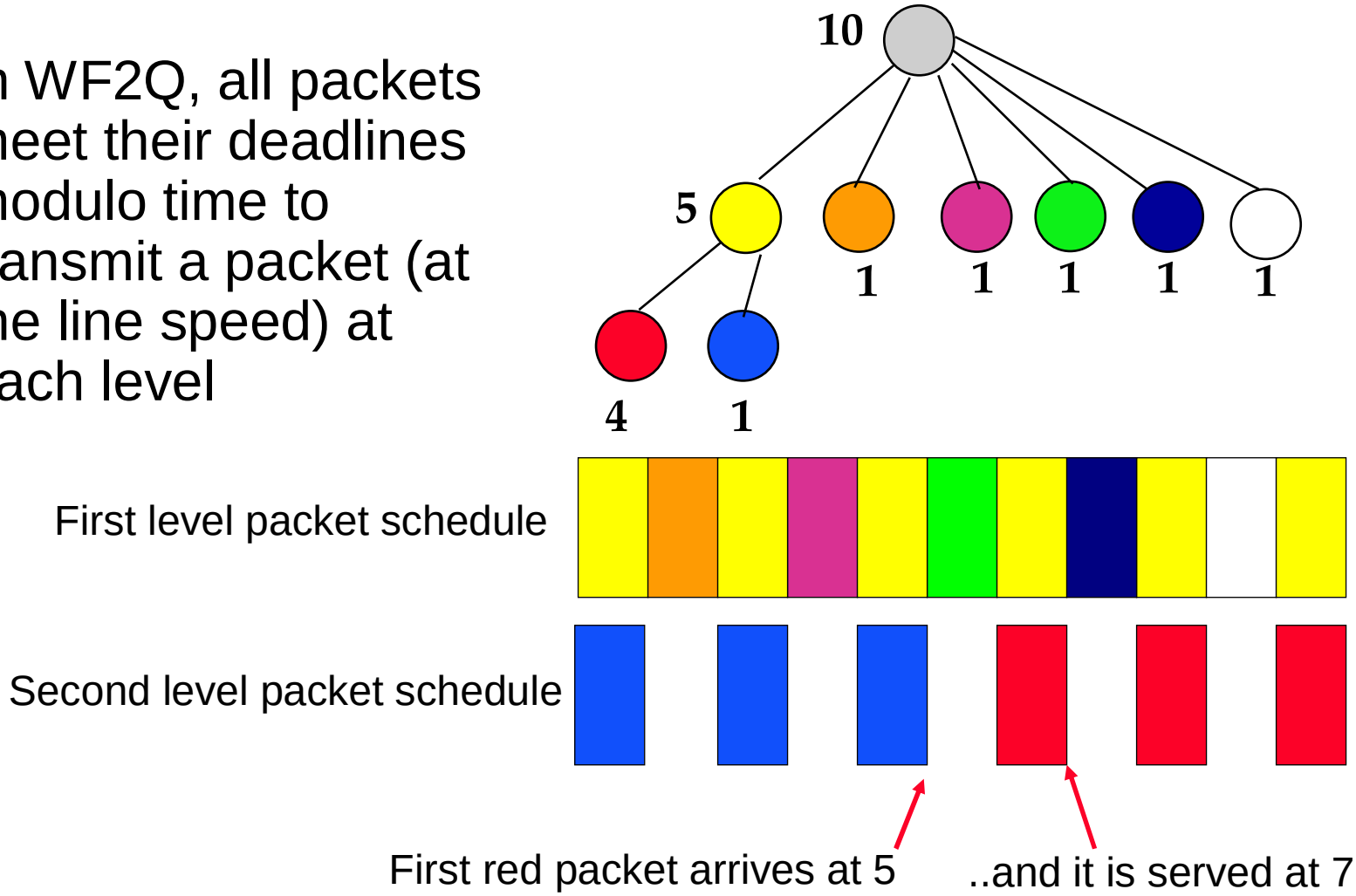
Second level packet schedule

First red packet arrives at 5

…but it is served at 11 !

Hierarchical-WF2Q Example

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



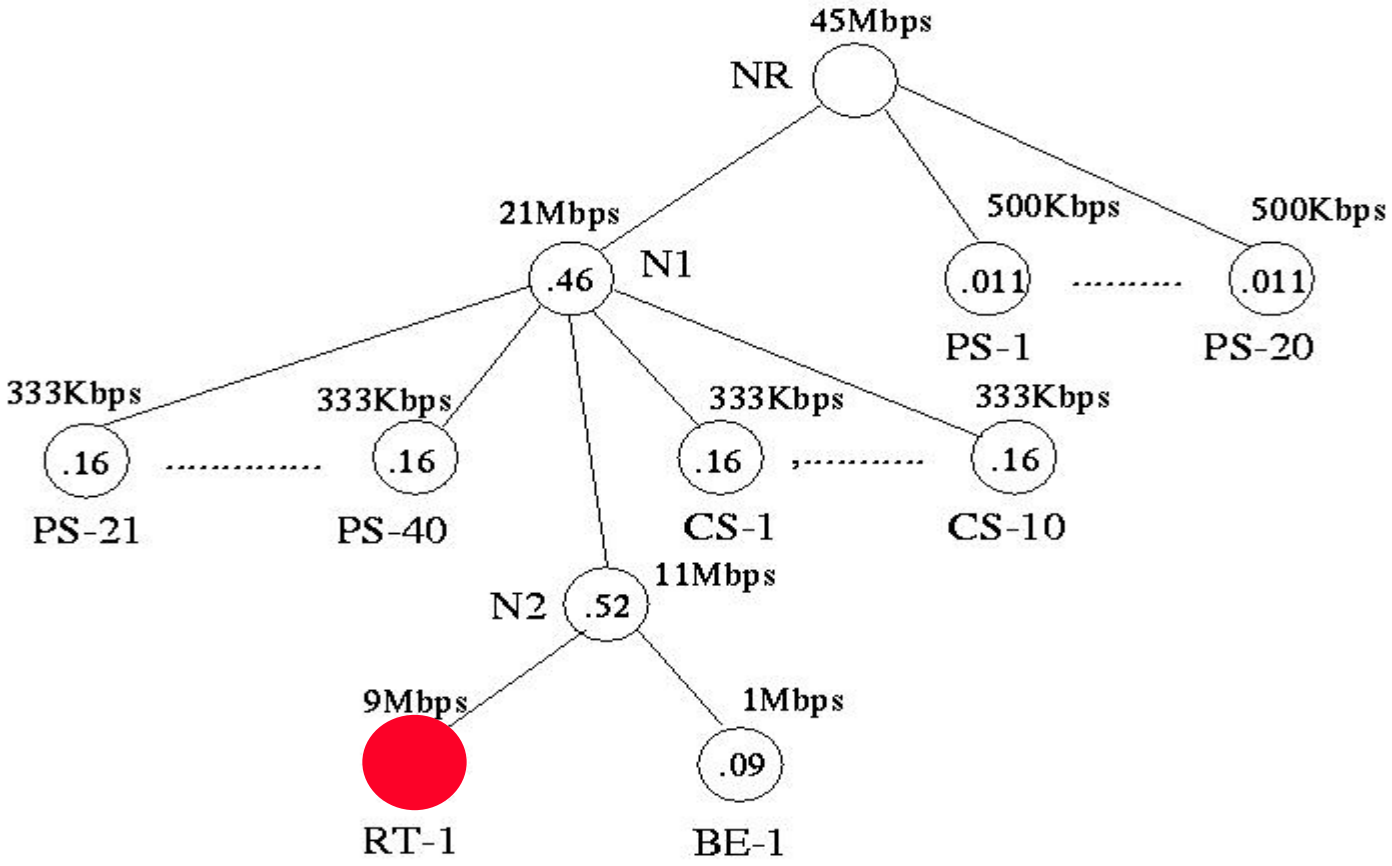
WF²Q+

- 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

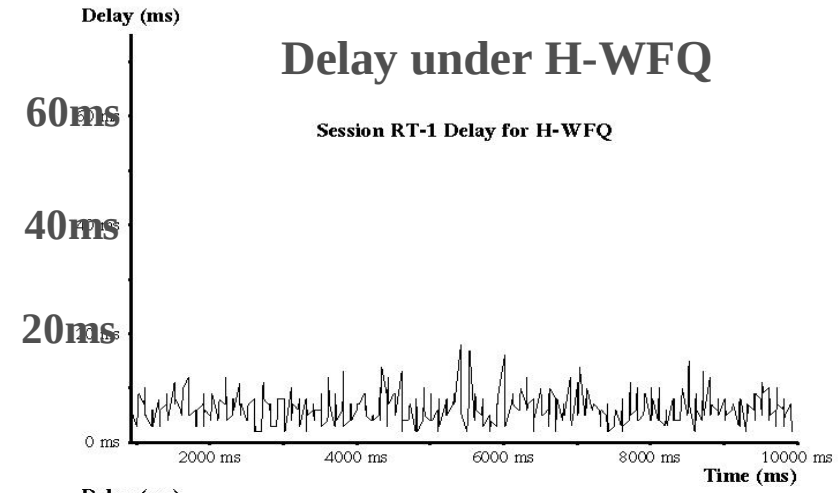
$$V_{WF^2Q+}(t+\tau) = \max(V_{WF^2Q+}(t) + W(t, t+\tau), \min_{i \in B(t+\tau)}(S_i^{h_i(t+\tau)}))$$

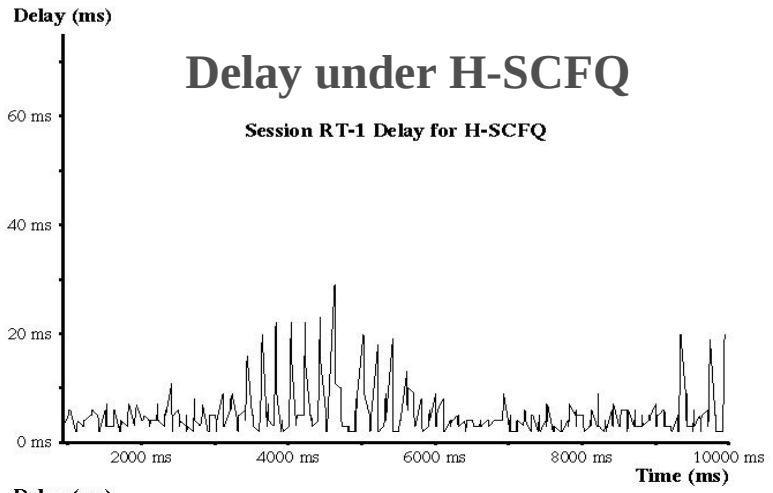
  - $h_i(t+\tau)$ - sequence number of the packet at the head of the queue of flow i
  - $S_i^{h_i(t+\tau)}$ - virtual starting time of the packet at the head of queue i
  - $B(t)$ - set of packets backlogged at time t in the packet system

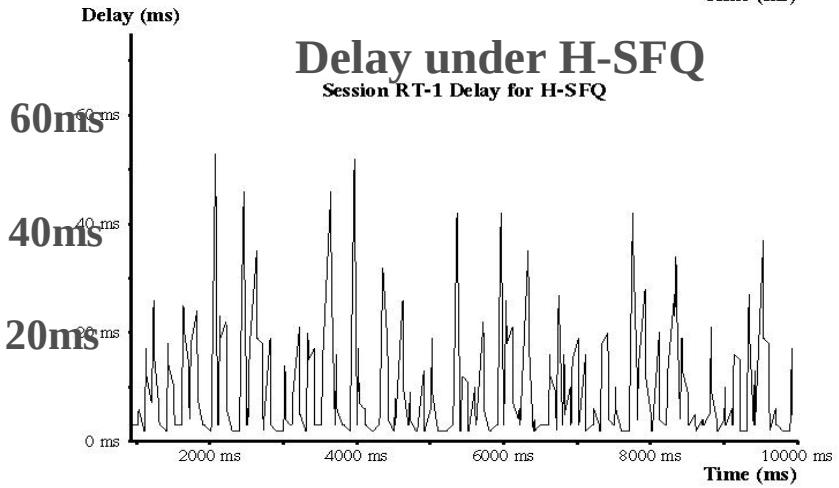
Example Hierarchy

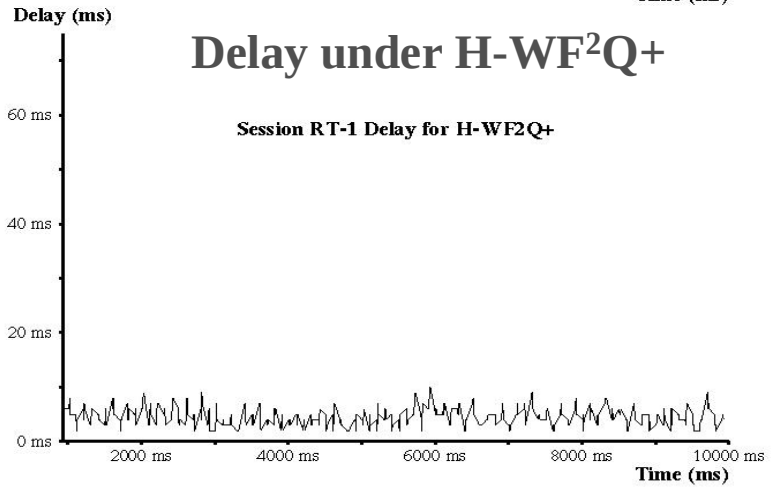


Uncorrelated Cross Traffic



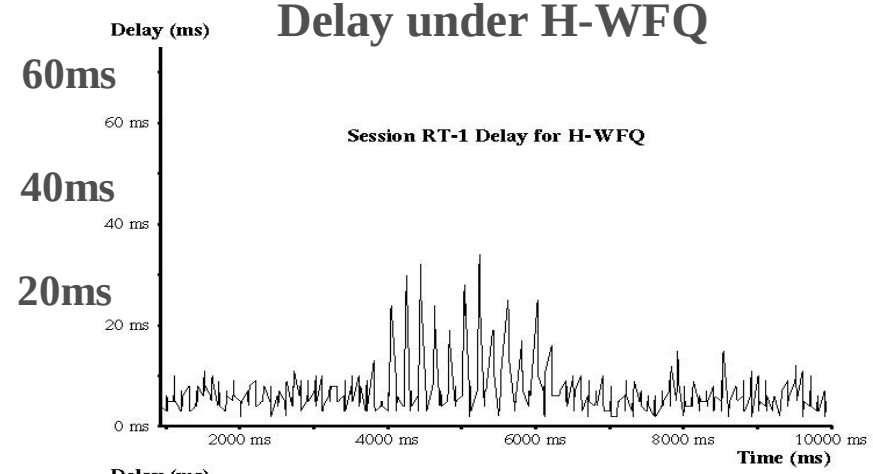




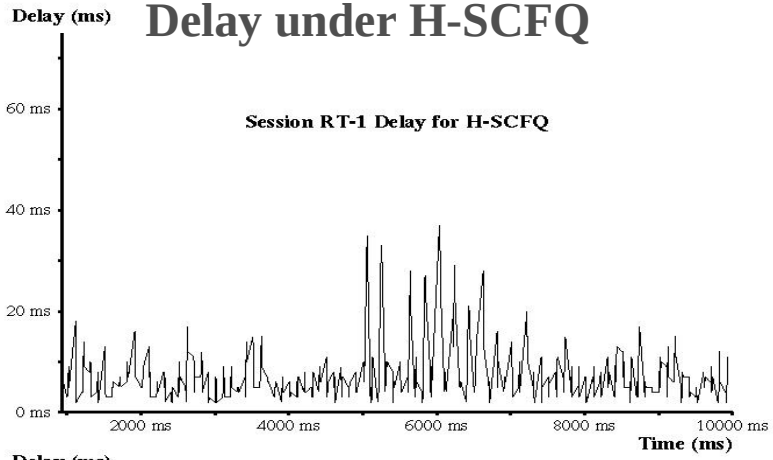


Correlated Cross Traffic

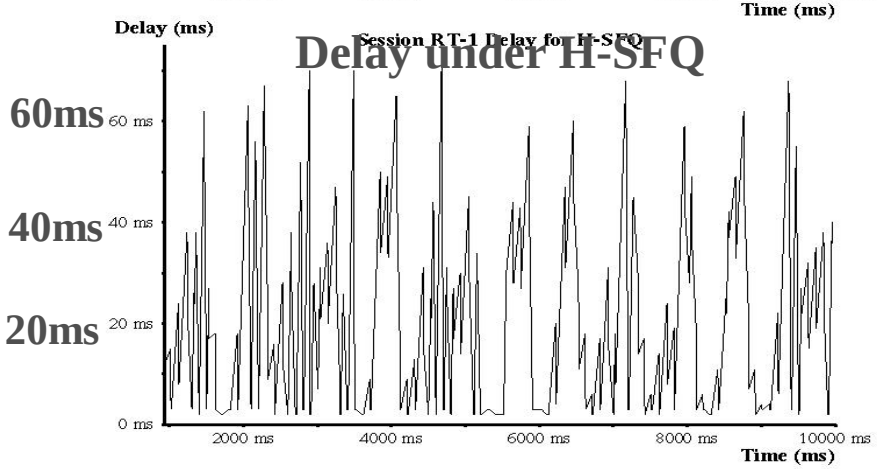
Delay under H-WFQ



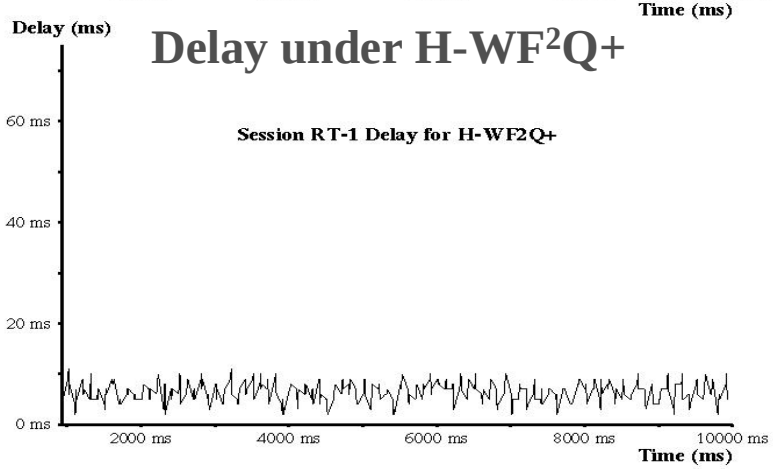
Delay under H-SCFQ



Delay under H-SFQ



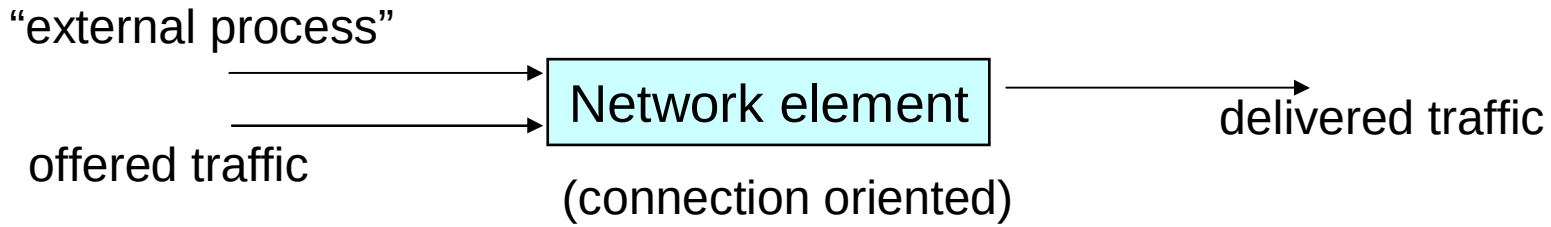
Delay under H-WF^{2}Q+



Why Service Curve?

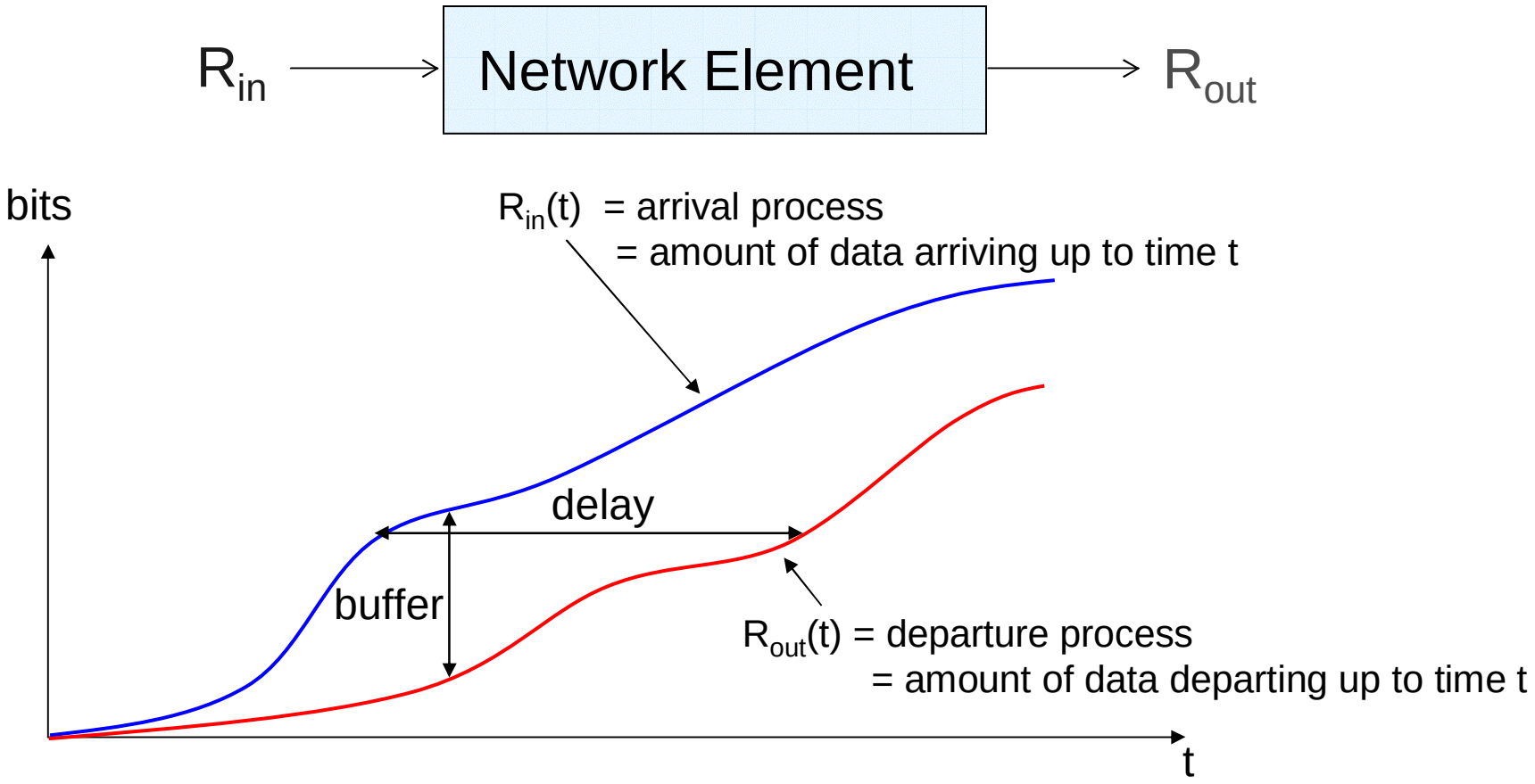
- WFQ, WF2Q, H-WF2Q+
  - Guarantee a minimum rate: $\geq C \times w_i / \sum_{j=1}^{N} w_j$
    - N – total number of flows
  - A packet is served no later than its finish time in GPS (H-GPS) modulo the sum of the maximum packet transmission time at each level
- For better resource utilization we need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

What is a Service Model?



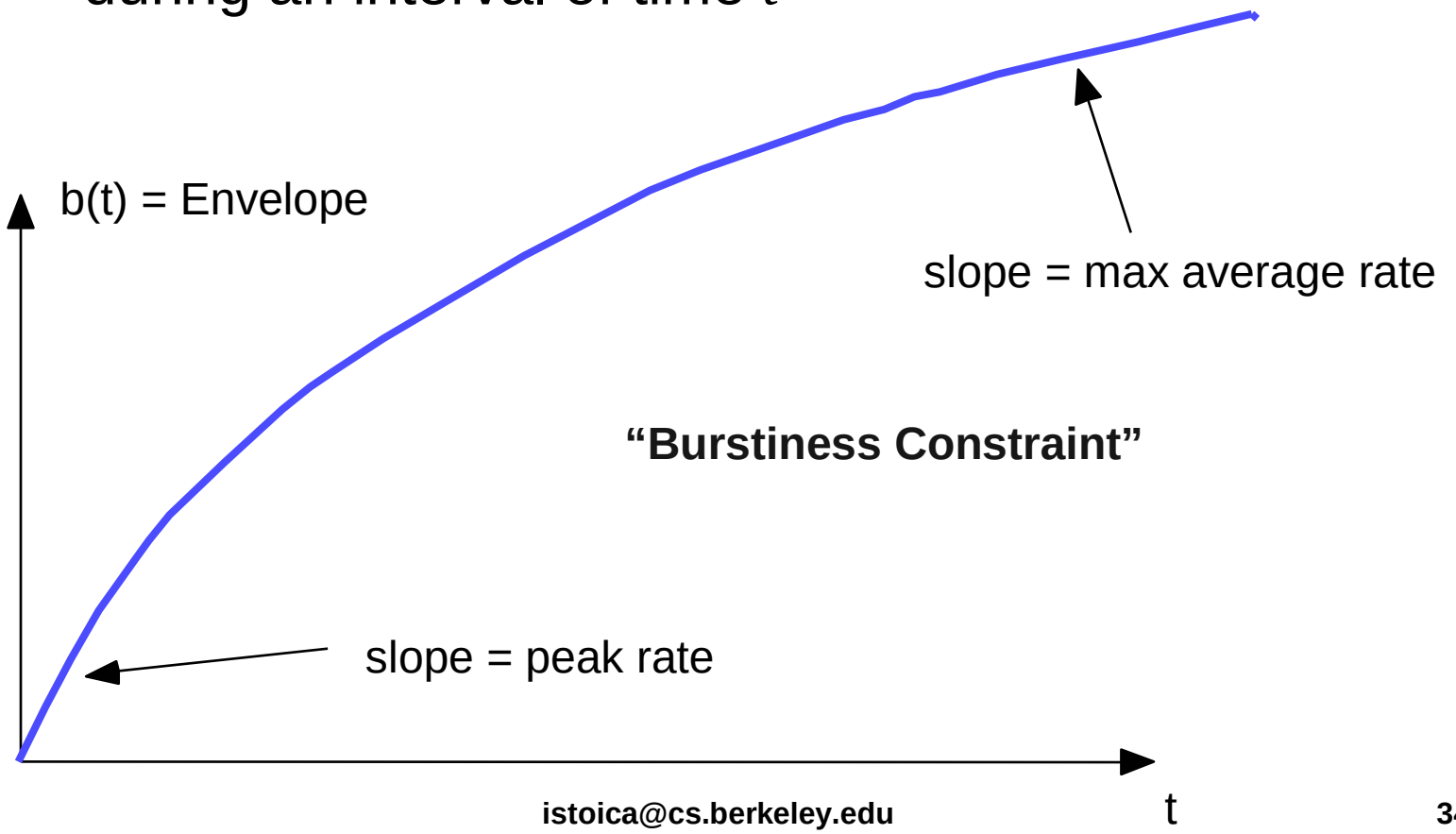
- The QoS measures (delay,throughput, loss, cost) depend on offered traffic, and possibly other external processes.
- A service model attempts to characterize the relationship between offered traffic, delivered traffic, and possibly other external processes.

Arrival and Departure Process



Traffic Envelope (Arrival Curve)

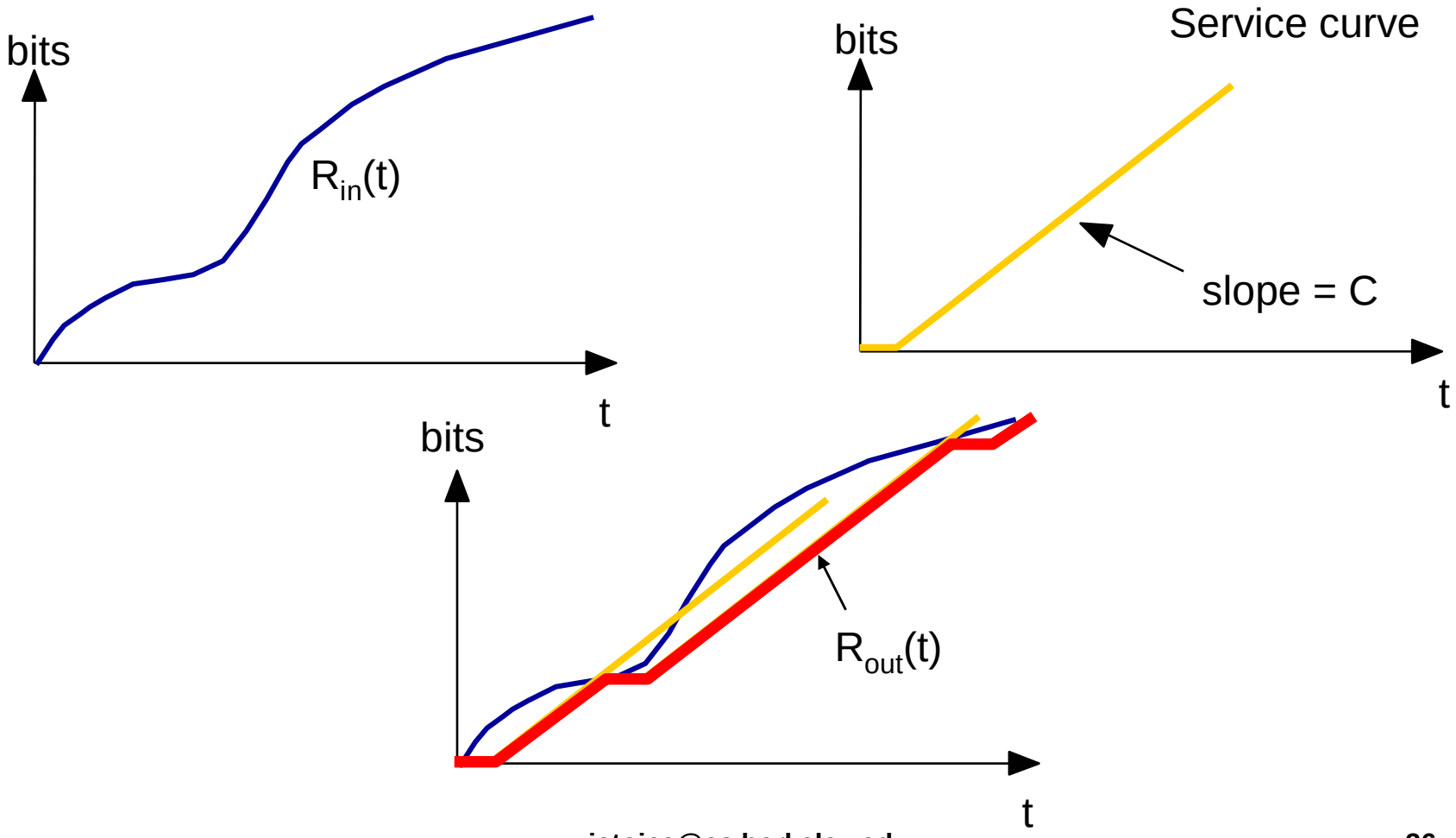
- Maximum amount of service that a flow can send during an interval of time t



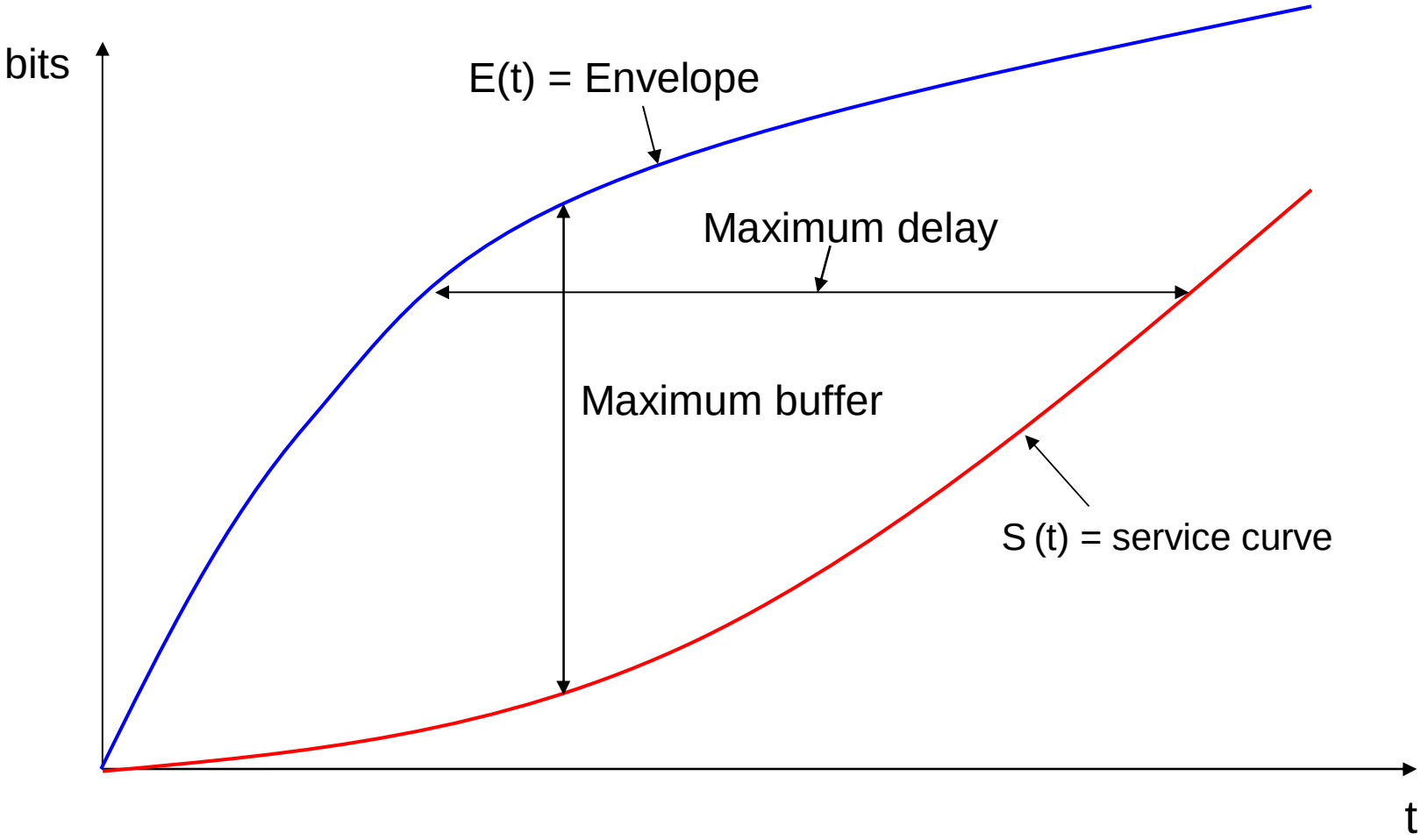
Service Curve

- 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)

Big Picture

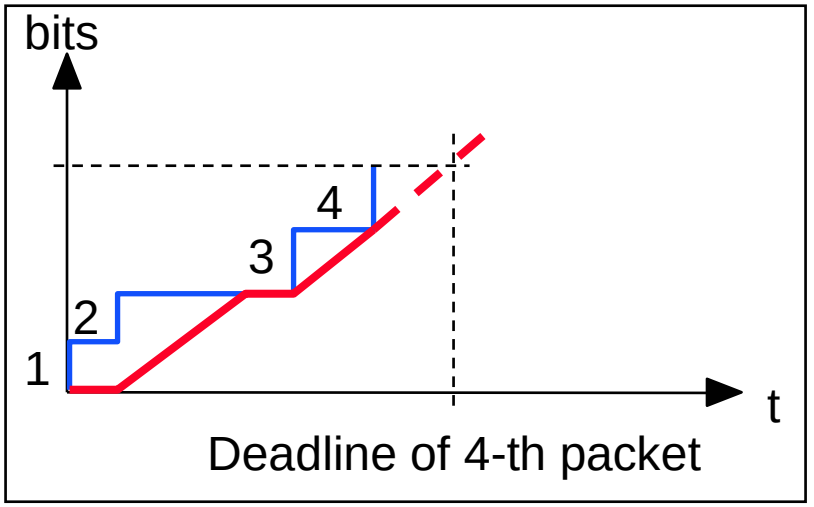


Delay and Buffer Bounds



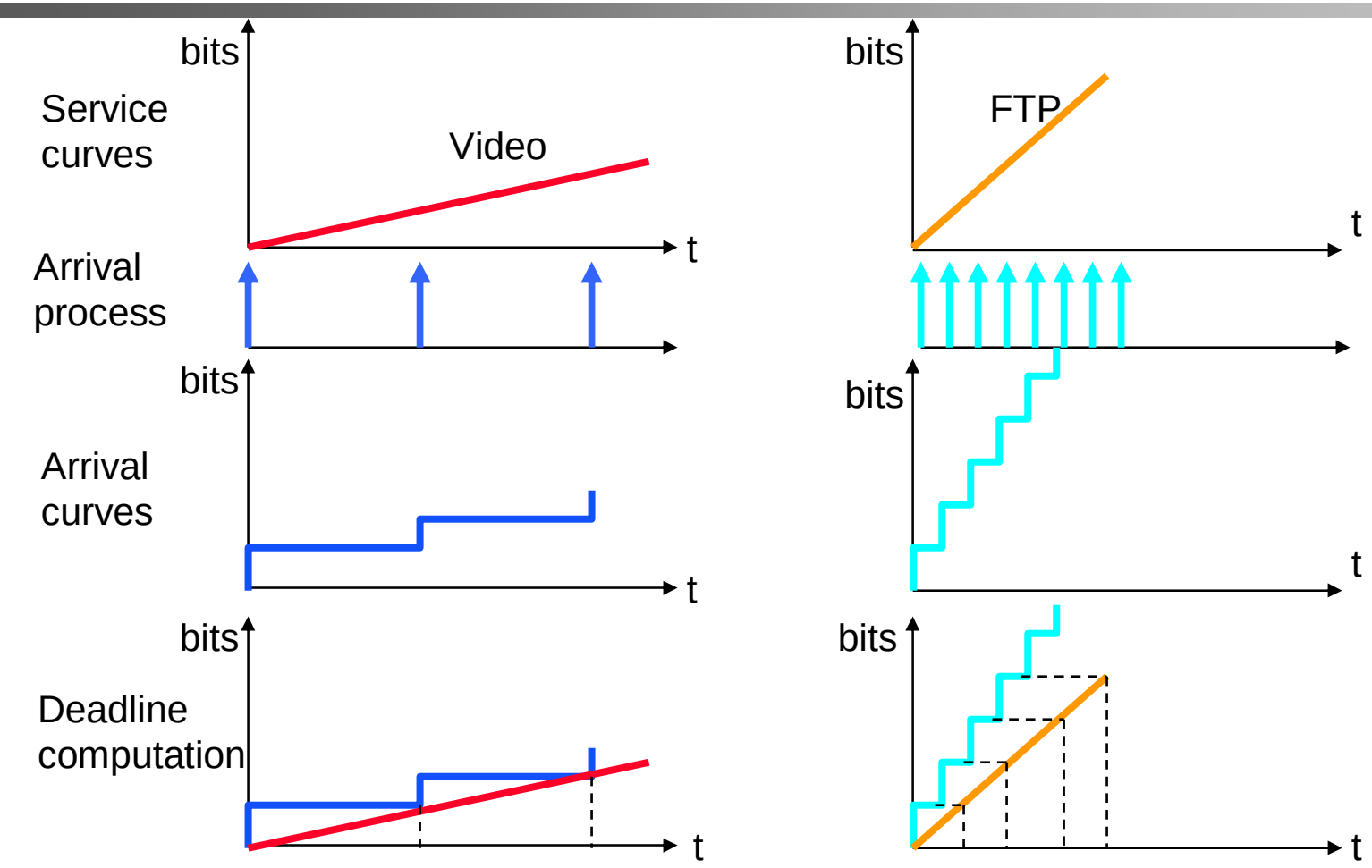
Service Curve-based Earliest Deadline (SCED)

- 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

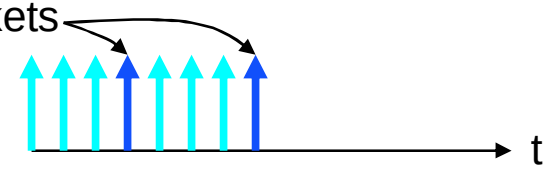


- Properties
  - If sum of all service curves <= $C*t$
  - All packets will meet their deadlines modulo the transmission time of the packet of maximum length, i.e., L_{max}/C

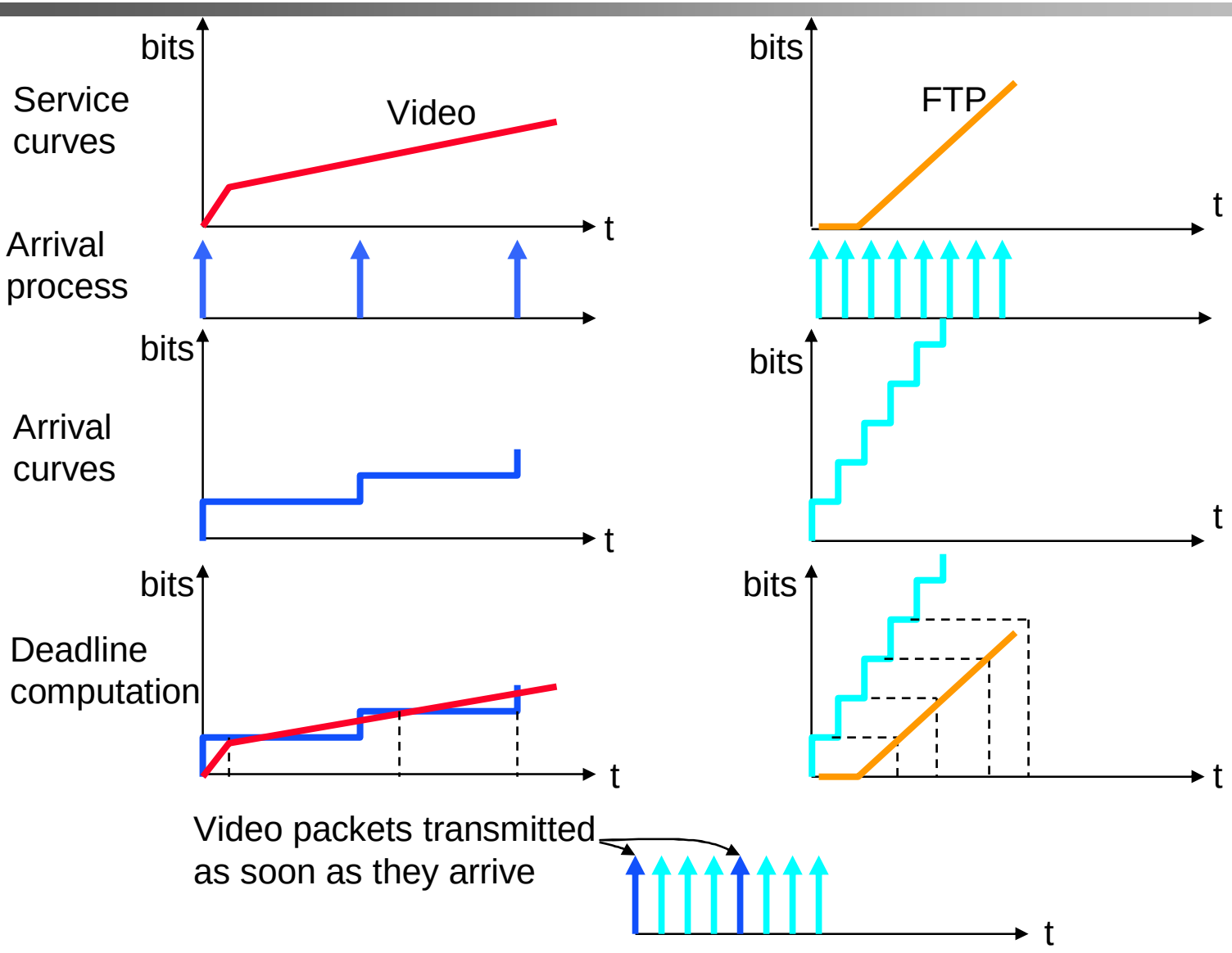
Linear Service Curves: Example



Video packets have to wait after ftp packets



Non-Linear Service Curves: Example



Summary

- WF2Q+ 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