Providing Hard Guarantees through Rate-Based and Rate-Controlled Service Disciplines

Victor Firoiu
Department of Computer Science
U. Massachusetts, Amherst

Networks Seminar, Department of ECE

Outline

- Introduction to Hard Guarantees: worst case analisys
- Analisys of Generalized Processor Sharing
- Analisys of Earliest Deadline First
- Comparison of GPS and EDF
- Conclusion

Hard vs Soft Guarantees

- Hard guarantees: bounds on worst case performance

  Ex: network sojourn time for all packets of a flow $\leq$ delay guarantee

- Soft guarantees: bounds on average performance

  Ex: $P(\text{netw. sojourn time} > d) < \epsilon_d$

- Soft Guarantees: Pros

  less conservative constraints $\Rightarrow$ higher network utilization

- Hard Guarantees: Pros

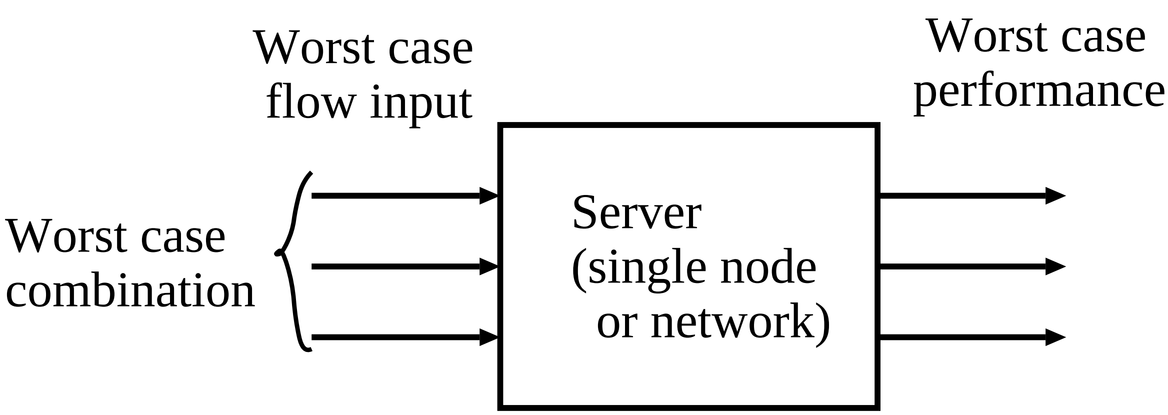
  clear determination of user-network contract violations

  worst cases might not be so improbable: e.g. flow correlations

Rest of the talk: on Hard Guarantees

The Framework of Providing Hard Guarantees

- Upper bounds on flow traffic
- Worst case of flow traffic combination
- Upper bound on the QoS provided to a flow



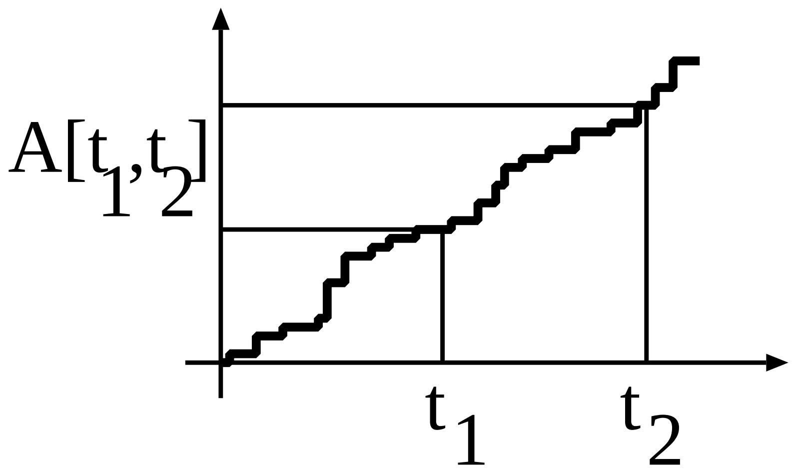
- For tight bounds:

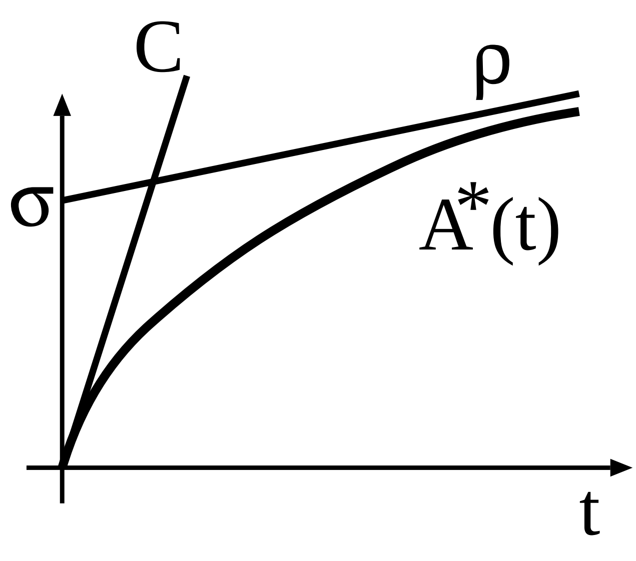
  least upper bounds on traffic

  $\Rightarrow$ achievable (least upper bound) on QoS

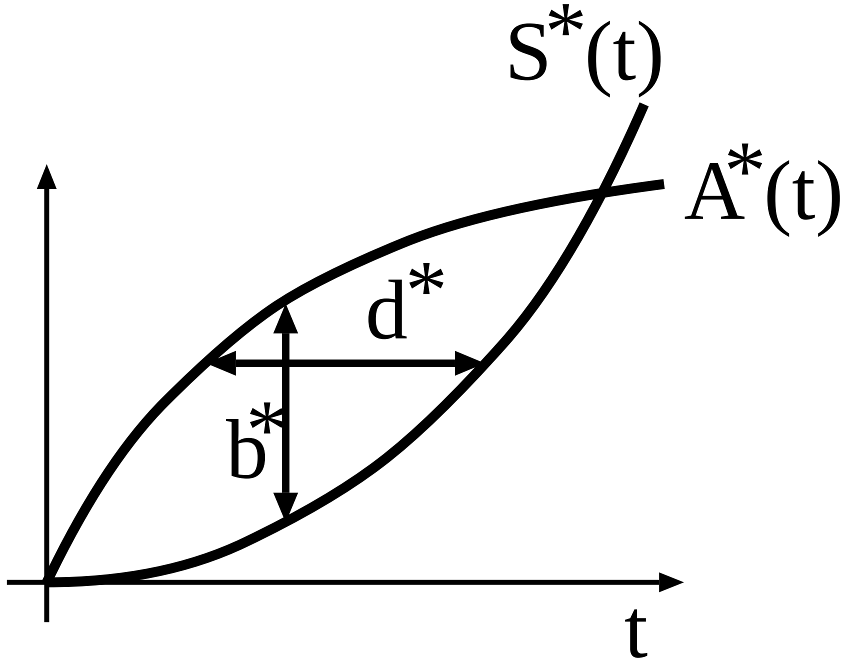
Traffic Envelopes: upper bounds on a flow's traffic

- Arrival function for a flow: $A[t_1, t_2]$
- Envelope characterization: $A[t, t+\tau] \leq A^*(\tau)$





Bounds on QoS Guarantees



- A^* - upper bound on arrivals
- S^* - lower bound on service
- d^* - upper bound on queuing delay
- b^* - upper bound on backlog

Motivation for Weigted Fair Queuing

- Protection against ill-behaved sources
  $\Rightarrow$ per-flow queuing
- Per-flow QoS (bw, delay) guarantees
  $\Rightarrow$ per-flow priority weights
- Service independent of packet lengths
  $\Rightarrow$ fluid flow model
- Fairness
  relevant for best-effort traffic
  not relevant for traffic with QoS guarantees

Generalized Processor Sharing

- Service definition:

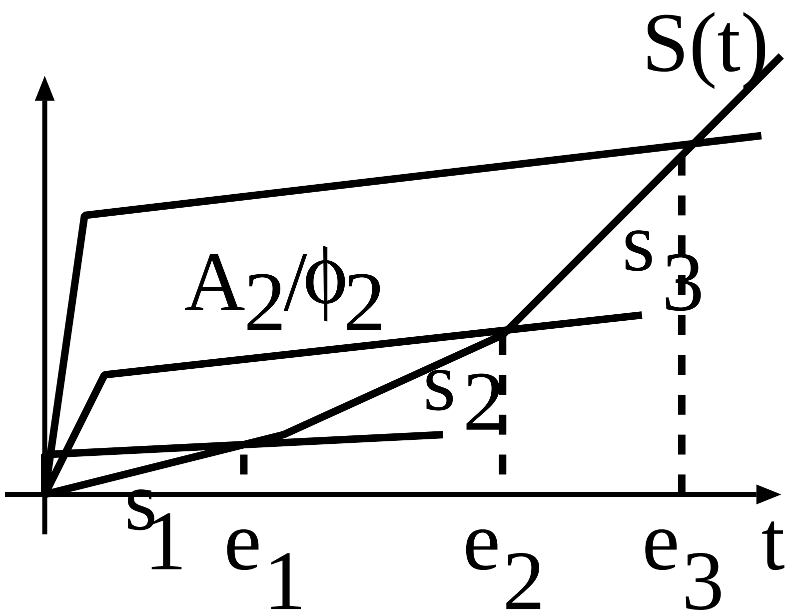
  for any backlogged flow i, $\frac{S_i[t_1,t_2]}{S_j[t_1,t_2]} \geq \frac{\phi_i}{\phi_j}, \quad \forall j \in \mathcal{N}$

  flow i is guaranteed a rate $R_i = \frac{\phi_i}{\sum_{j \in \mathcal{N}} \phi_j} c$

  Obs: can get more

- Single node calculus

  worst case achieved by "All-greedy regime"



$S(t)$, Universal Service Curve

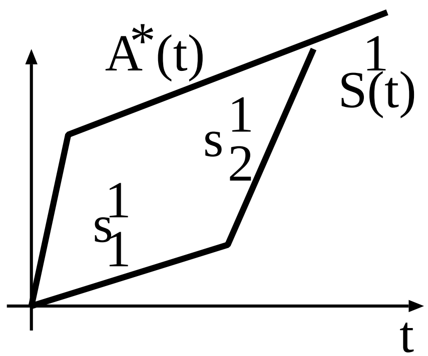
$e_k \stackrel{\Delta}{=}$ end of flow k's busy period

$s_k = \frac{s_{ik}}{\phi_i} = \frac{c - \sum_{j=1}^{k-1} \rho_j}{\sum_{j=k}^{N} \phi_k}, \quad \forall i \geq k$

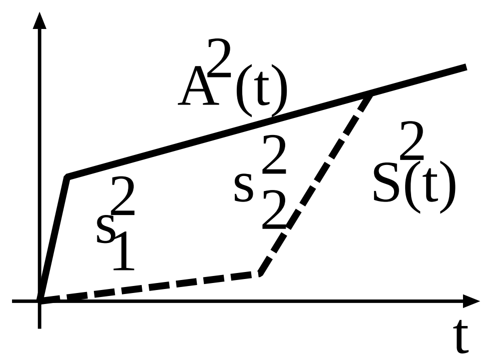
GPS: Multiple Node Calculus

- Worst case: staggered greedy regime

Node 1, flow i service curve

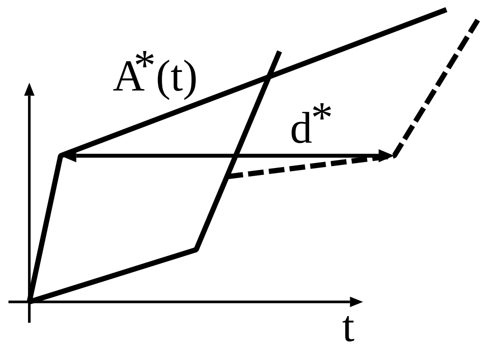


Node 2, flow i service curve

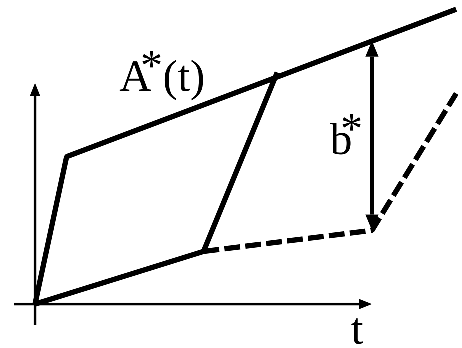


GPS: Multiple Node Calculus (cont'd)

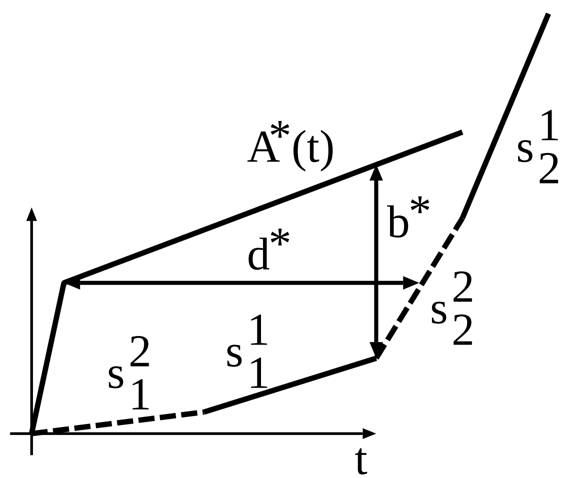
Staggered greedy regime for delay bound



Staggered greedy regime for buffer bound



- Network service curve for flow i $\quad s_1^1 < s_1^2 < s_2^1 < s_2^2$



Simplified Cases of GPS

- Problems with the GPS model

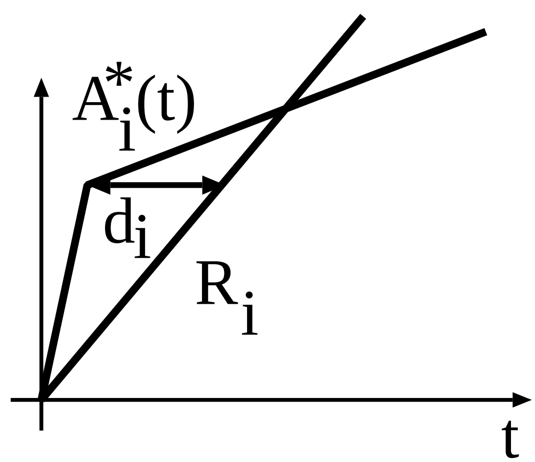
  hard to compute ϕ_i given d_i

  recompute all ϕ_i when a flow joins or leaves

  recompute all ϕ_i in all nodes

- A simplified model (usu. referred to as "WFQ")

  guarantee rate R_i



if all nodes guarantee R_i,

then d_i is an end-to-end delay guarantee

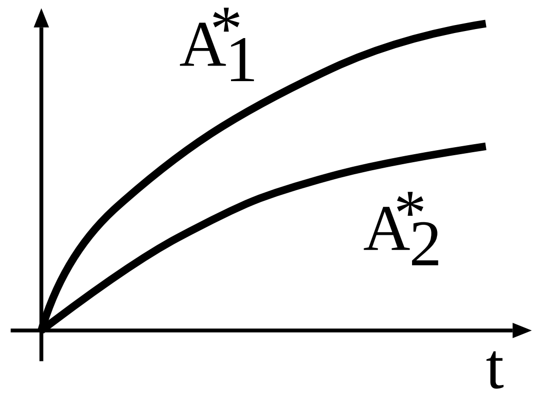
Obs: server is "virtually" non-work conserving

Intuition for EDF schedulability

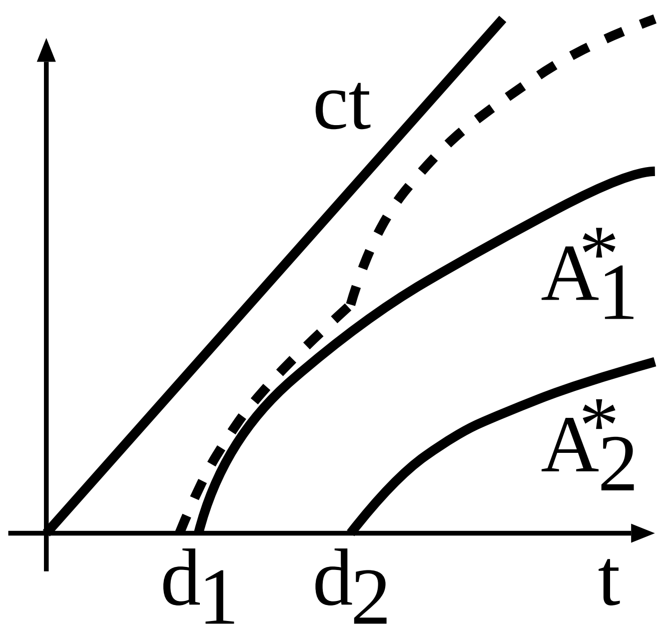
- EDF schedulability for two flows:

$$A_1^*(t-d_1) + A_2^*(t-d_2) \le ct$$

- worst case: greedy regime



- most delayed work $\le$ link work capacity



An EDF admission control algorithm

- Problem

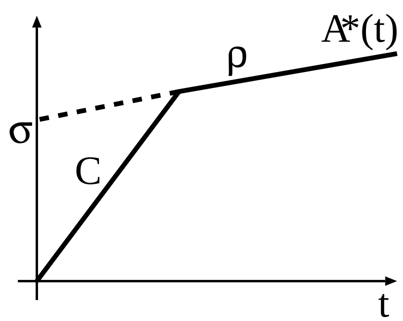
  Given a set of flows $(A_i^*, d_i)_{i \in \mathcal{N}}$

  Compute <u>fast</u>:

  given A_f^*, what is minimum delay $\overline{d}_f$

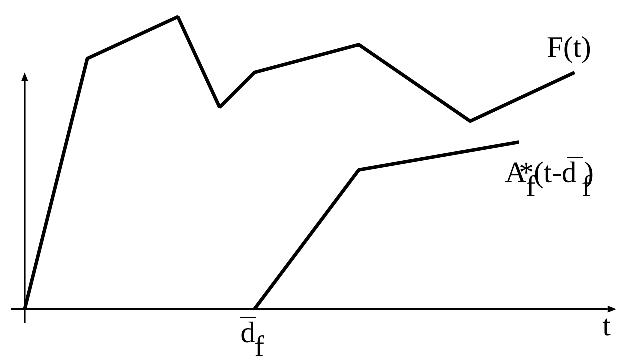
  reserve resources for (A_f^*, d_f)

- Assumption: two-segment, (C, σ, ρ) envelopes



- The exact solution: define $F(t) = ct - \sum_{i \in \mathcal{N}} A_i^*(t - d_i)$

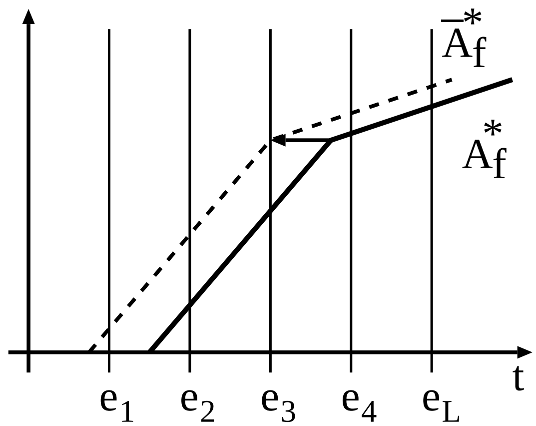
  check all N convex points of F



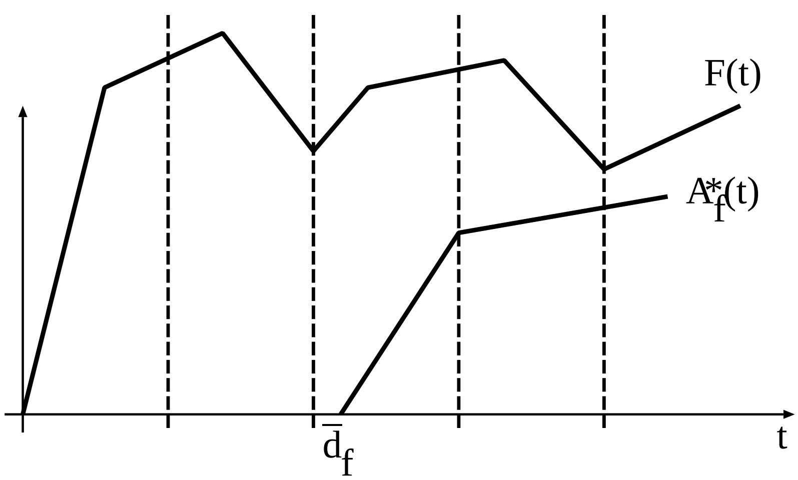
- Problem: algorithm complexity $O(N)$ depends on no of flows

A class-based EDF admission control algorithm

- Key idea: flex points can only take values from a small set $\mathcal{L}$



- Result: an $O(L)$ admission control algorithm



EDF/WFQ Comparison: Single Node

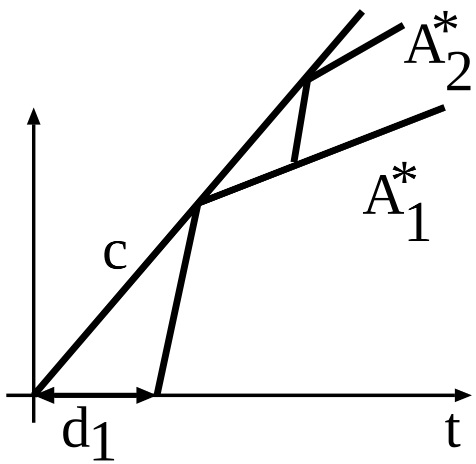
- EDF proven to be optimal:

  any set of tasks schedulable under any policy,

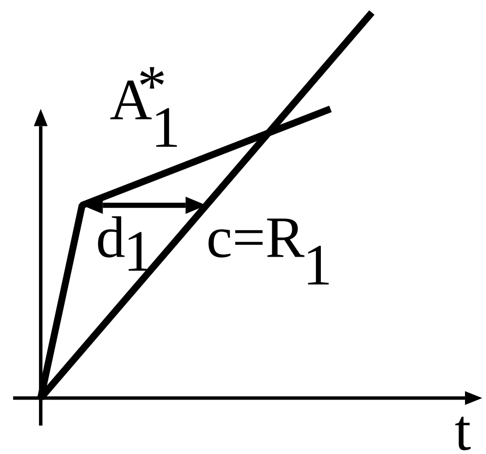
  ⇒ schedulable under EDF

- Example:

EDF



WFQ



$R_1 = c \Rightarrow$ no more service available for A_2

EDF/WFQ Comparison: Network Case

- Problem with naive Rate-Controlled EDF

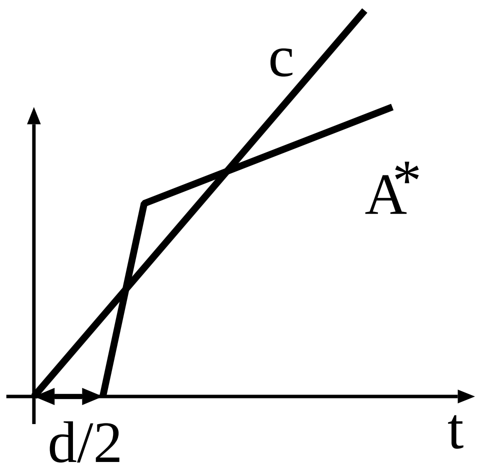
  EDF: e-e delay = $\sum$ (local delays)

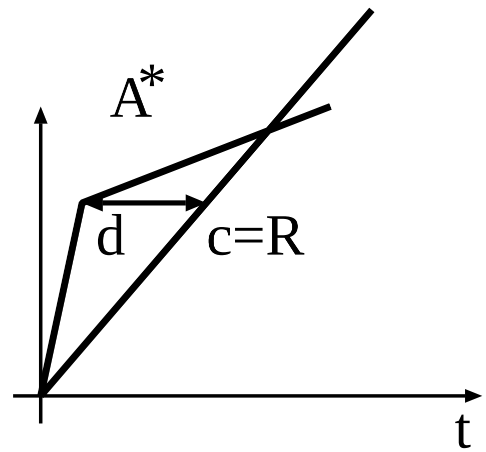
  WFQ: e-e delay = max(local delays)

  $\Rightarrow$ for the same e-e delay, WFQ reserves less than EDF

- Example: two nodes

  EDF: unschedulable at both nodes WFQ: schedulable at both nodes



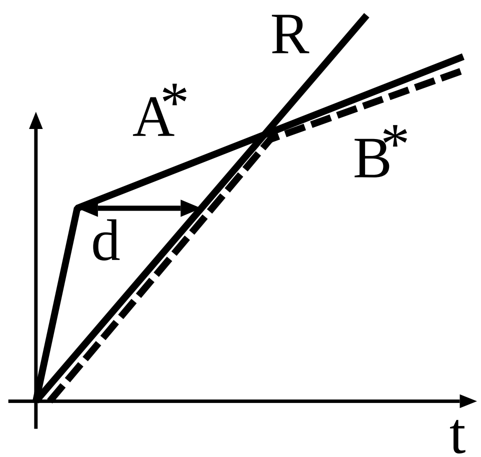


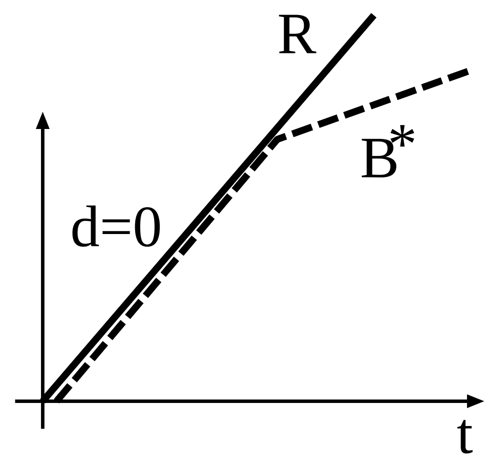
EDF/WFQ Comparison: Network Case (cont'd)

- Guarantee same e-e delay with Rate-Controlled WFQ:

  reshape (smooth) flow at rate R

  guarantee $d = 0$ at each node





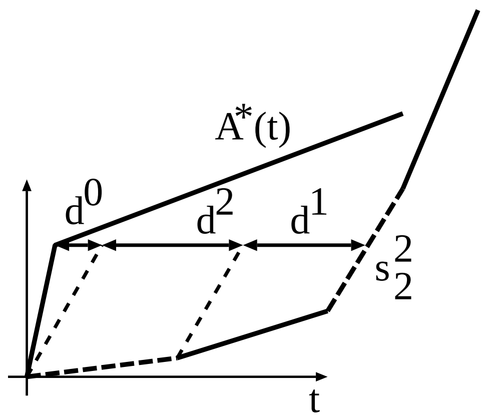
- Can replace WFQ with EDF at each node ($d = 0$)

  $\Rightarrow$ network-EDF at least as good as network-WFQ

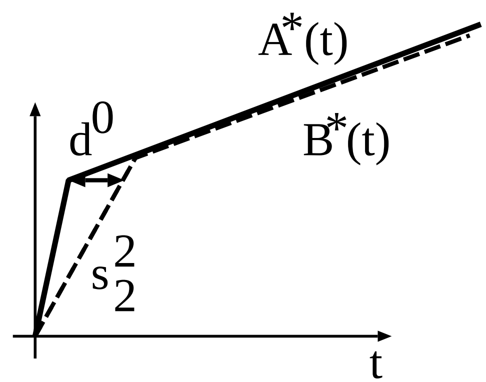
EDF/GEneral GPS Network Comparison

- Guarantee same e-e delay with Rate-controlled GPS

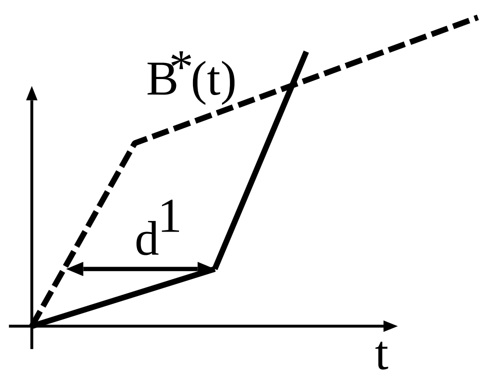
Network service curve



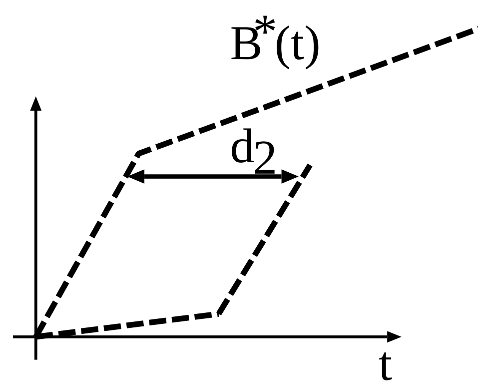
Delay in shaper



Delay in node 1



Delay in node 2

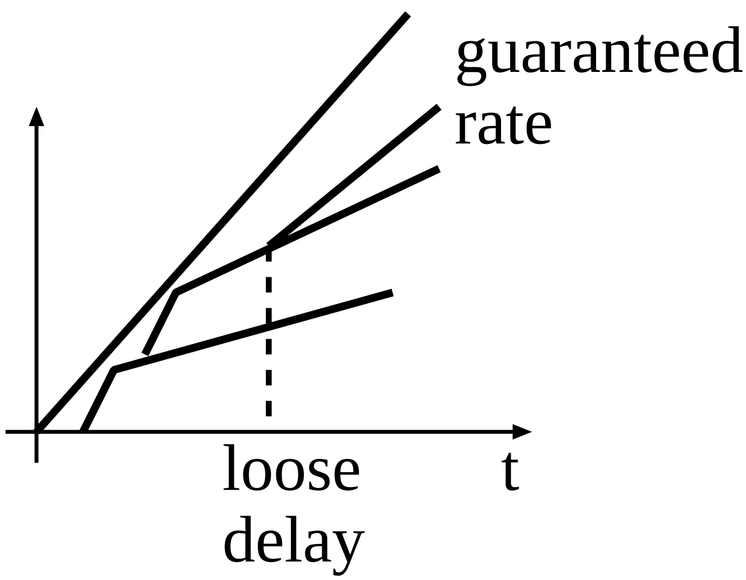


Parctical utility of EDF

- Besides flows with delay guarantees

  can “squeeze” flows with rate guarantees

  and loose delay requirements



Conclusion

- Hard Guarantees: a reliable user-network contract
- GPS - complicated admission control in the general case
- WFQ - has simplified admission control
- EDF - has reasonably simple admission control
- EDF - outperforms GPS in theory and practical applications