ÉCOLE POLYTECHNIQUE
FÉDÉRALE DE LAUSANNE

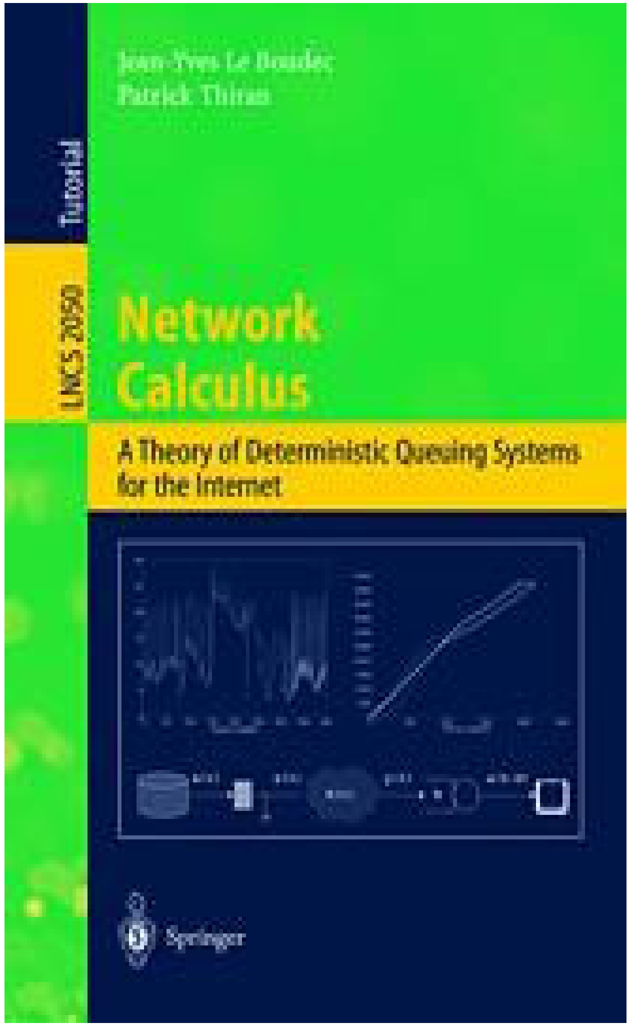
Network calculus :

Jean-Yves Le Boudec and Patrick Thiran
LCA-ISC, I&C, EPFL

CH-1015 Lausanne
Jean-Yves.Leboudec Patrick.Thiran @epfl.ch
http://lcawww.epfl.ch

Contents

1. Greedy shapers and arrival curves, min-plus convolution
2. Service curves, backlog, delay bounds
3. Diffserv: intuition and formal definition behind EF
4. Min-plus algebra in action: Video smoothing
5. Statistical multiplexing with EF

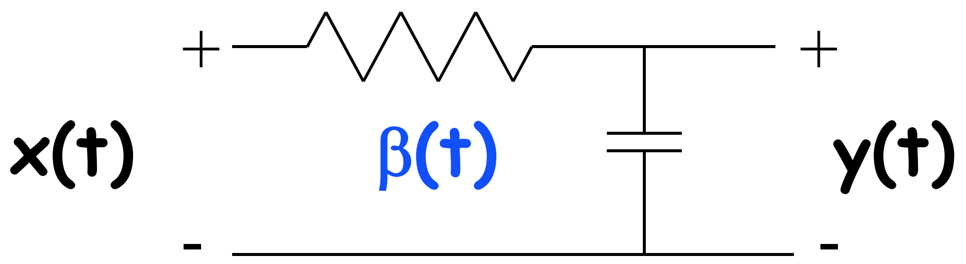


free on-line at :
http://lcawww.epfl.ch

What is Network Calculus ?

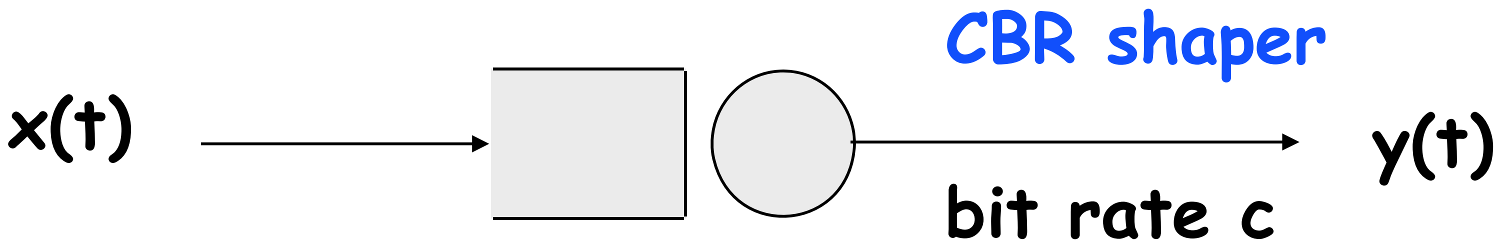
- Deterministic analysis of queuing / flow systems arising in communication networks
- Uses Min-Plus, Max-Plus and sometimes Min-Max algebra

The standard Linear Theory



- A LTI filter in conventional algebra (R, +, ×)
  - Input signal = electrical voltage $x(t)$
  - System = circuit (filter) with impulse response $\beta(t)$
  - Output = convolution of $x(t)$ and $\beta(t)$:
    $y(t) = \int \beta(t-s)\, x(s)\, ds$

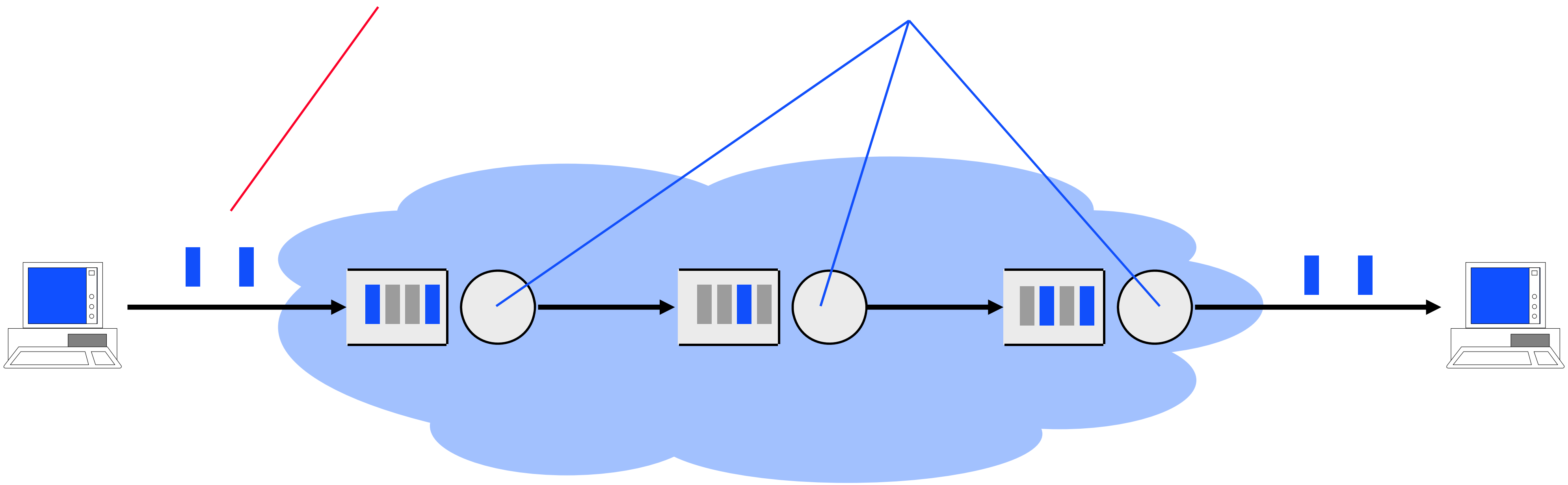
Network Calculus uses Min-Plus Linear Theory



- A linear system in min-plus algebra (R, min, +)
  - Input = arrived traffic in $[0,t]$, $x(t)$
  - System = CBR trunk of rate c : $\beta(t) = ct$
  - Output = convolution of $x(t)$ and $\beta(t)$:
    $y(t) = \inf_s \{\beta(t-s) + x(s)\}$

Two key Concepts
Arrival and Service Curves

- IntServ and DiffServ use the concepts of *arrival curve* and *service curves*



Contents

1. Arrival curves
- Arrival curve: definition
- Leaky bucket and GCRA
- Arrival curve and min-plus convolution
- Good arrival curves are sub-additive
- Minimal arrival curve and min-plus deconvolution
- Greedy shaper and its properties
- Packetization

2. Service curves, GPS, backlog, delay bounds

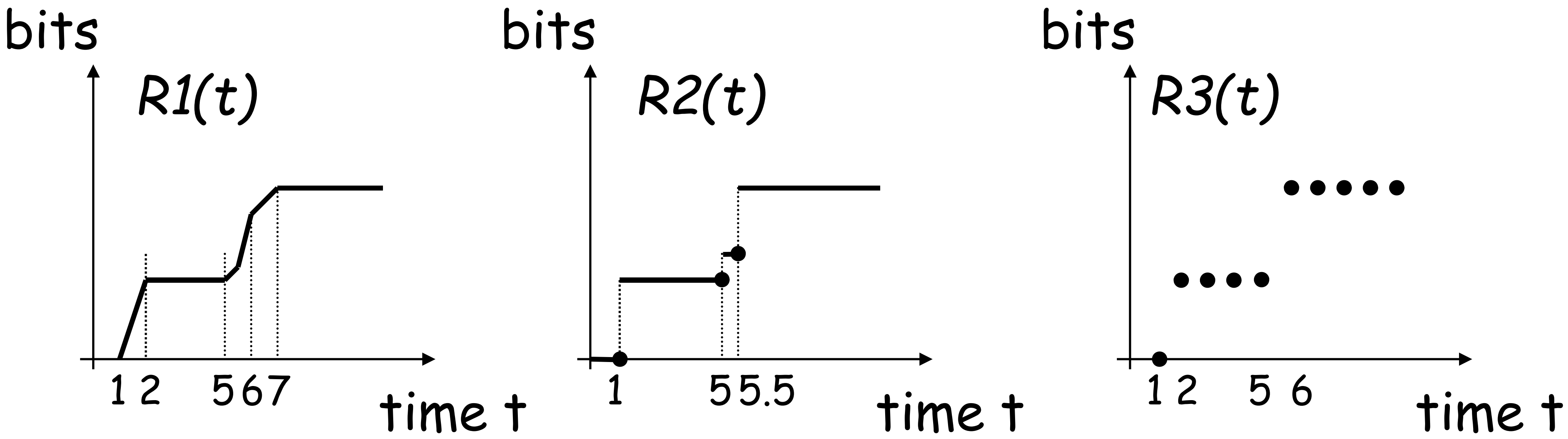
3. Diffserv: intuition and formal definition behind EF

4. Min-plus algebra in action: Video smoothing

5. Statistical multiplexing with EF

Cumulative flows

- Cumulative flow $R(t) \in \mathcal{F}$, t real or integer
- $\mathcal{F} = \{\, x(t) \mid x(t) \text{ is non decreasing and } x(t) = 0 \text{ for } t < 0 \,\}$
- **Examples:**



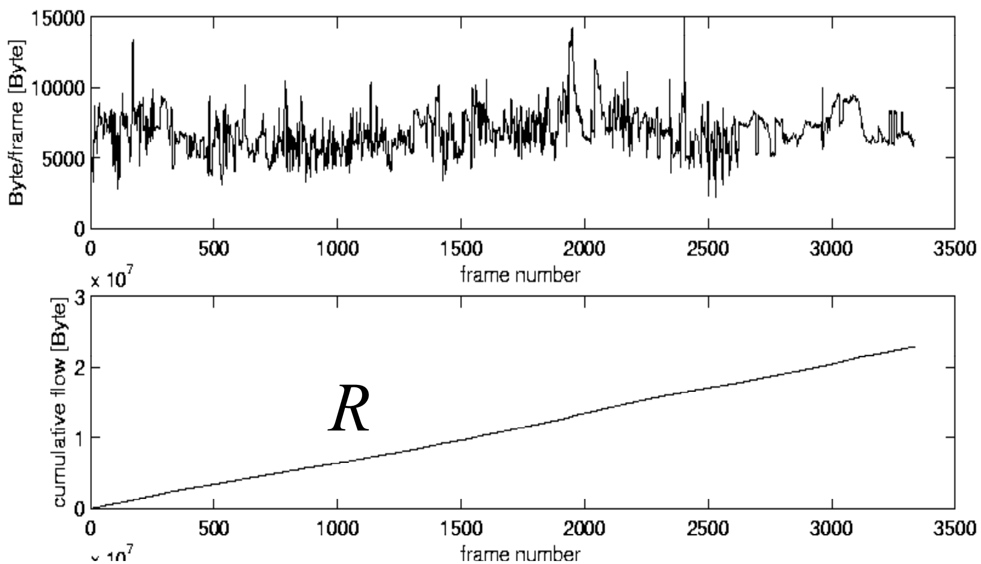
Fluid model (continuous)

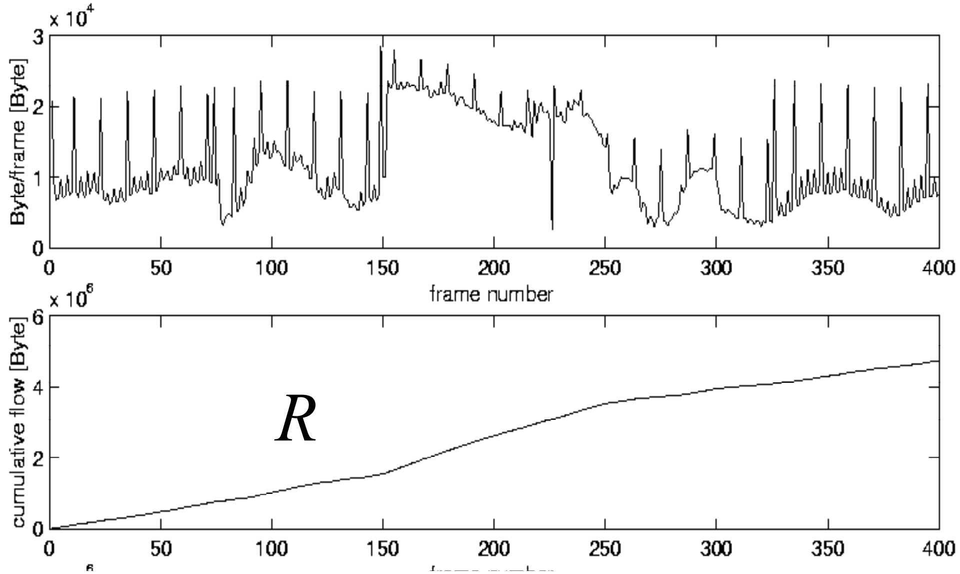
Packet model (left continuous)

Discrete-time model (left continuous)

Example

- MPEG files, 25 frames/sec





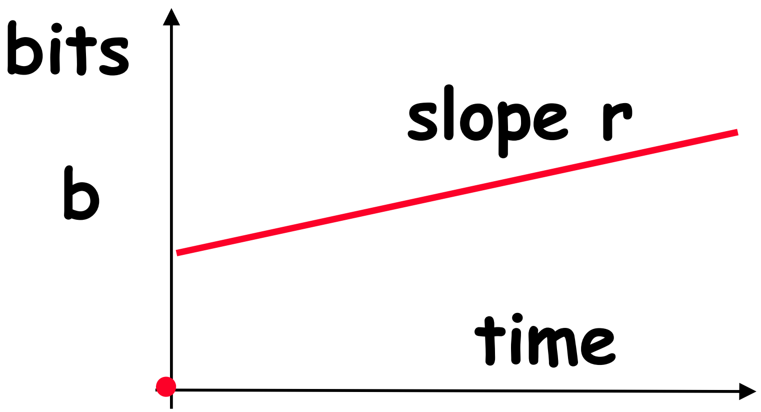
Arrival Curves

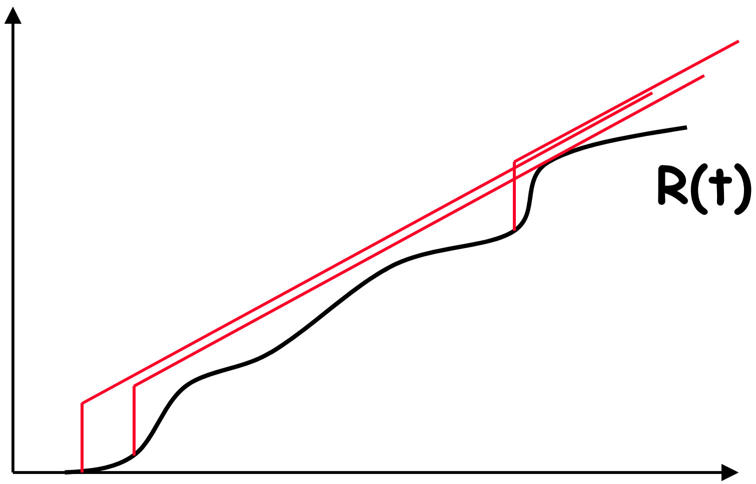
- Arrival curve α: For any times $0 \le s \le t$, the cumulative flow R(.) satisfies

$$R(t) - R(s) \le \alpha(t-s)$$

Example 1: affine arrival curve $\gamma_{r,b}$

$$\alpha(t) = \gamma_{r,b}(t) = rt+b \quad \text{for } t>0$$





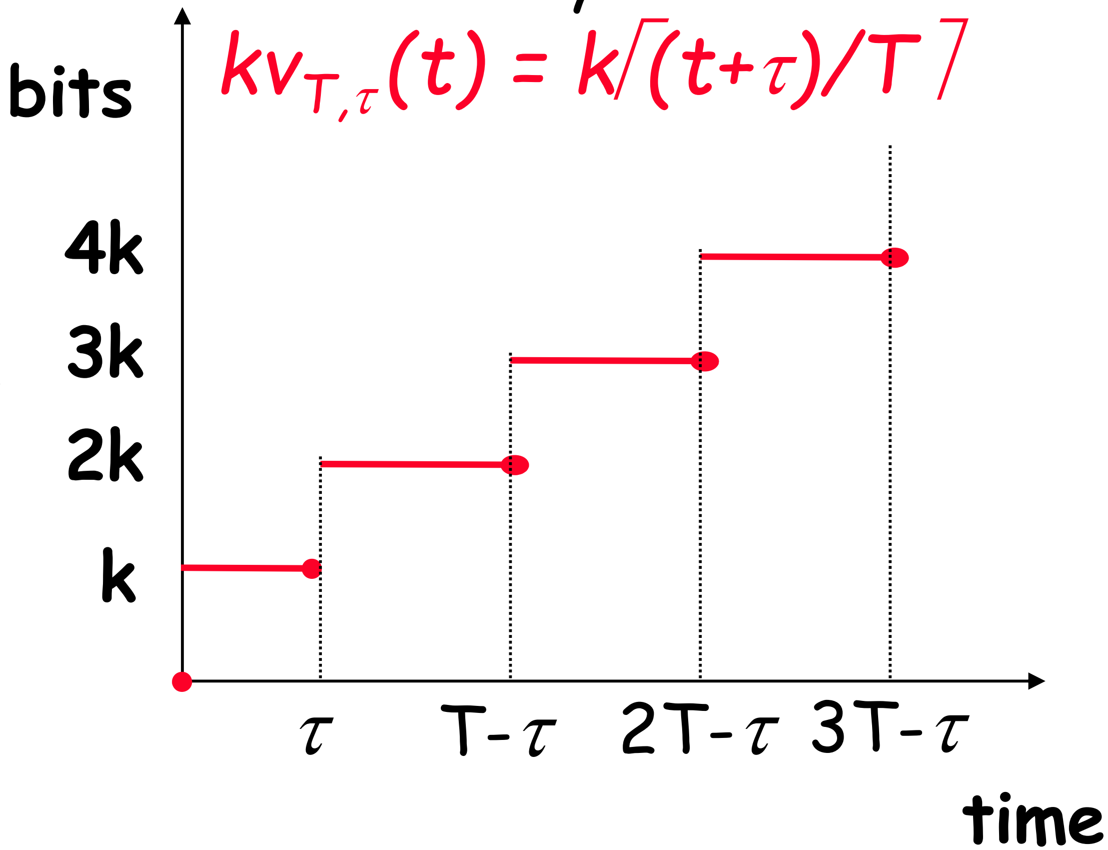
Arrival Curves

Example 2: stair arrival curve $kv_{T,\tau}$

- $\alpha(t) = kv_{T,\tau}(t) = k \lceil (t+\tau)/T \rceil$ with T = period, τ = tolerance, k = constant packet size
- Characterizes flows that are periodic stream of packets of same size k (cells), that suffers a variable delay <= τ

- All packets of size k. Then R conforms to $\alpha = kv_{T,\tau}$

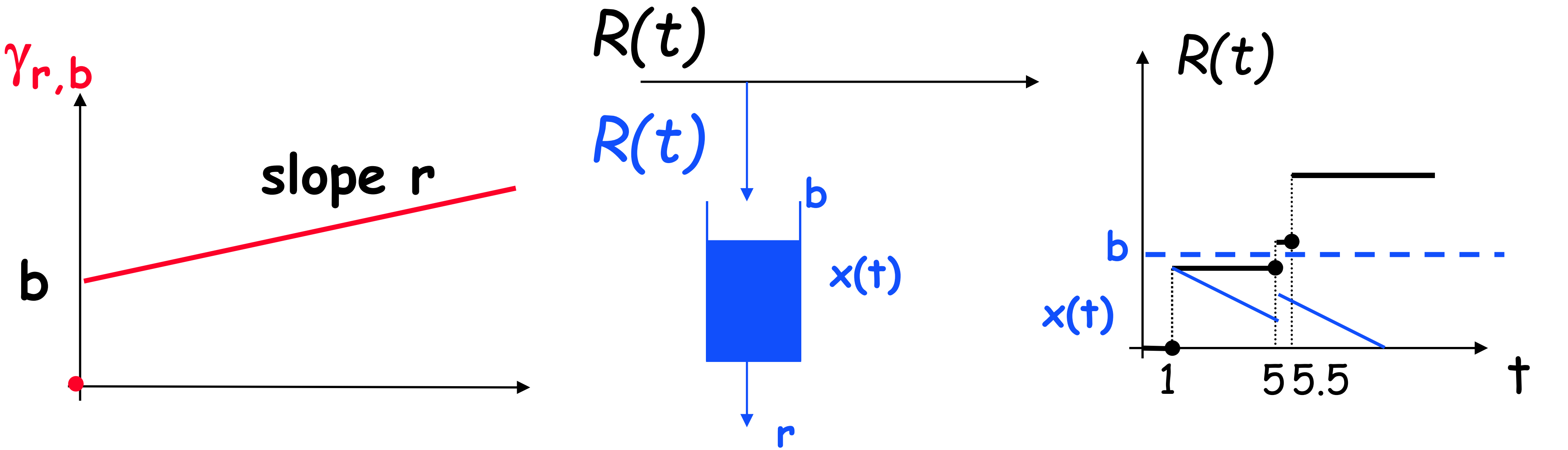
$\Leftrightarrow$ R conforms to $\alpha = \gamma_{r,b}$ with $r = k/T$ and $b = k(\tau+T)/T$



Leaky bucket

- All packets of flow R are declared conformant by a leaky buket controller of rate r and size b

⬄ R conforms to $\alpha(t) = \gamma_{r,b}(t) = rt+b$ for $t>0$



GCRA (T, τ)

- All packets (cells) of flow R are of the same size k
- Arrival time of nth = A_n
- Theoretical arrival just after nth arrival is $\theta_n = \max(A_n, \theta_{n-1}) + T$
- If $A_{n+1} >= \theta_n - \tau$ then cell is conformant, otherwise not

Example: GCRA (10,2)

n	1	2	3	3	4	5
θ_{n-1}	0	11	21	21	31	41
A_n	1	11	16	20	29	38
	c	c	nc	c	c	nc

- Equivalences: R conforms to GCRA (T,τ)

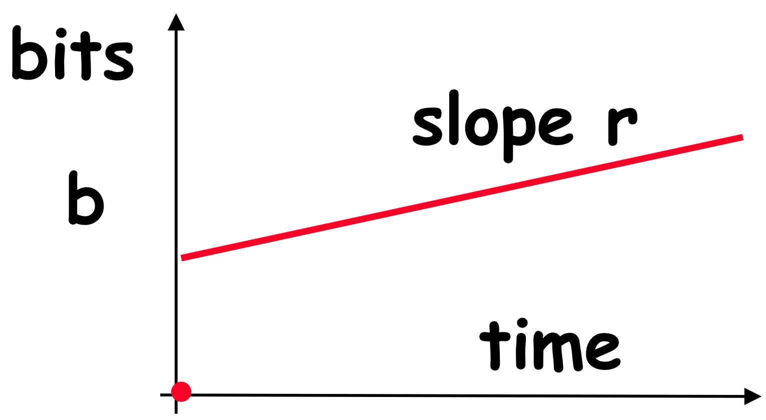
$\Leftrightarrow$ R conforms to staircase arrival curve $\alpha = k v_{T,\tau}$

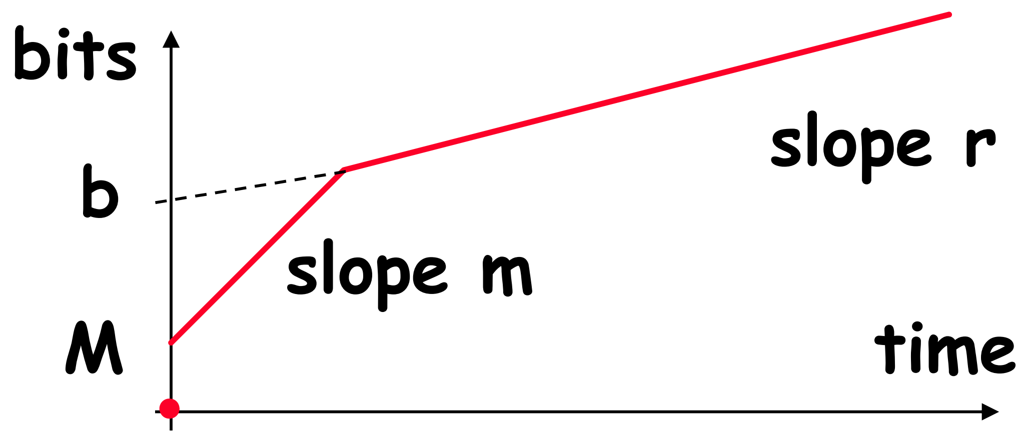
$\Leftrightarrow$ R conforms to leaky bucket $(r = k/T,\ b = k(\tau+T)/T)$

$\Leftrightarrow$ R conforms to affine arrival curve $\alpha = \gamma_{r,b}$

Combining leaky buckets

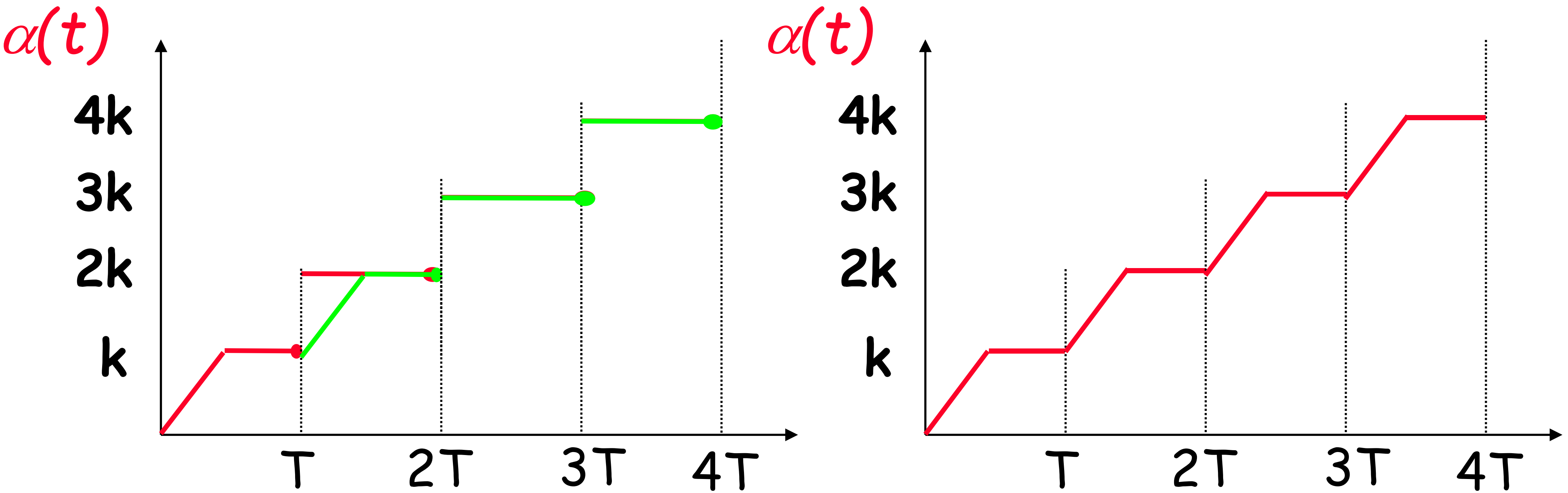
- standard arrival curve in the Internet ($\wedge$ = min)
  $\alpha(u) = \min(pu+M, ru+b) = (pu+M) \wedge (ru+b)$





Sub-additivity and arrival curves

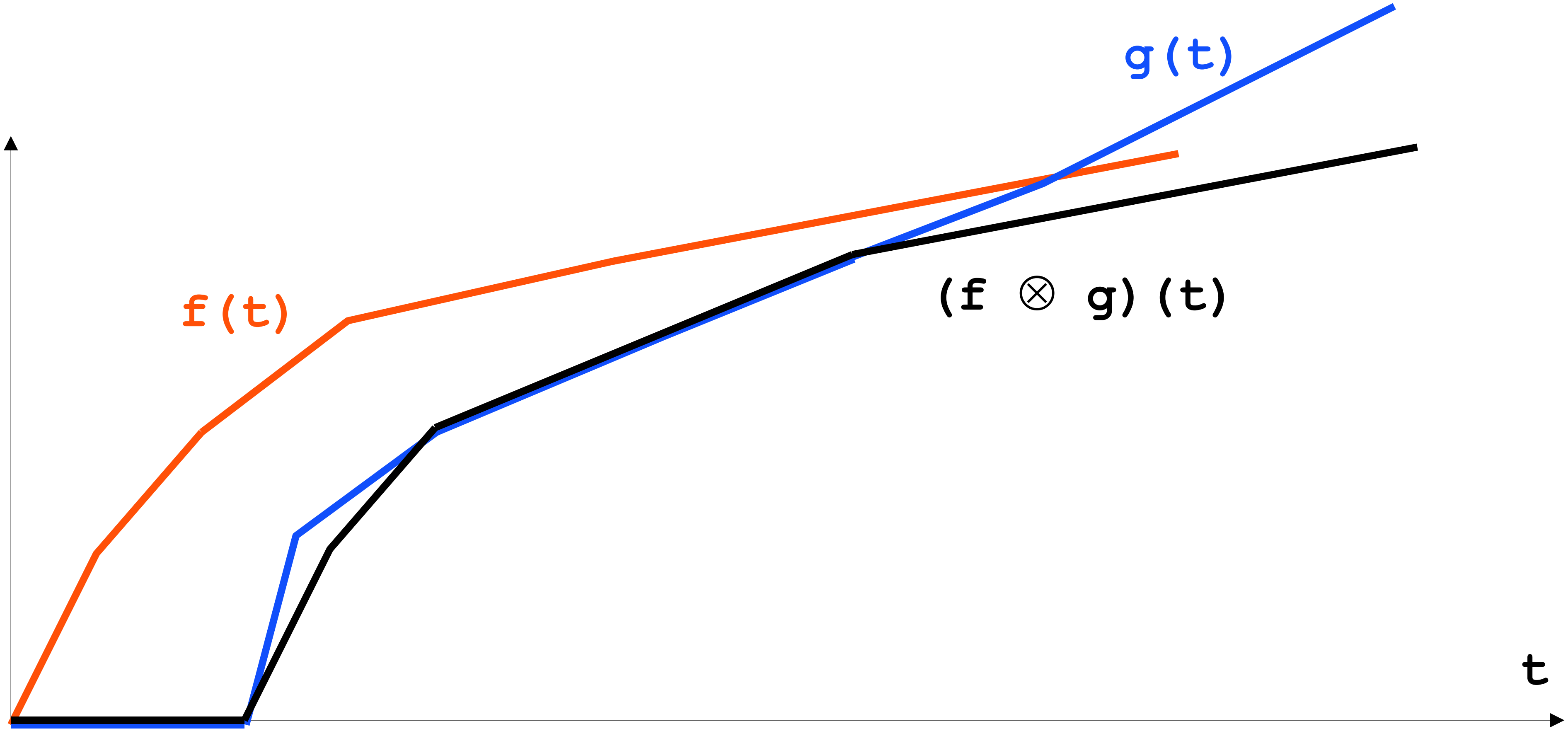
- If α is an arrival curve for flow R, so is $\bar{\alpha}$
- $\bar{\alpha}(t) \leq \alpha(t)$
- *What is* $\bar{\alpha}(t)$ *?*
- The answer uses min-plus convolution and sub-additivity



Min-plus convolution ⊗

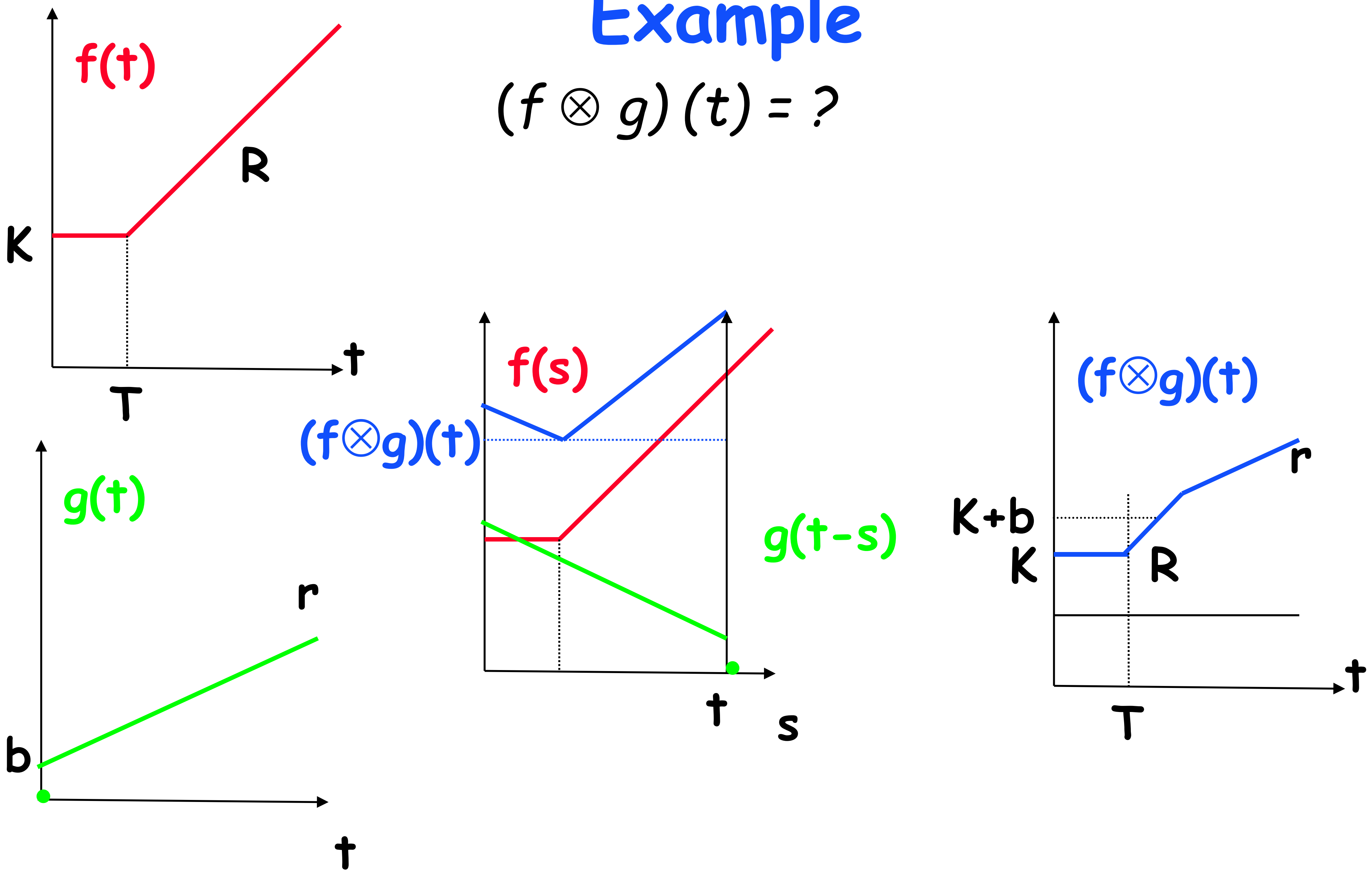
- Definition

$$(f \otimes g)(t) = \inf_u \{ f(t-u) + g(u) \}$$



Example

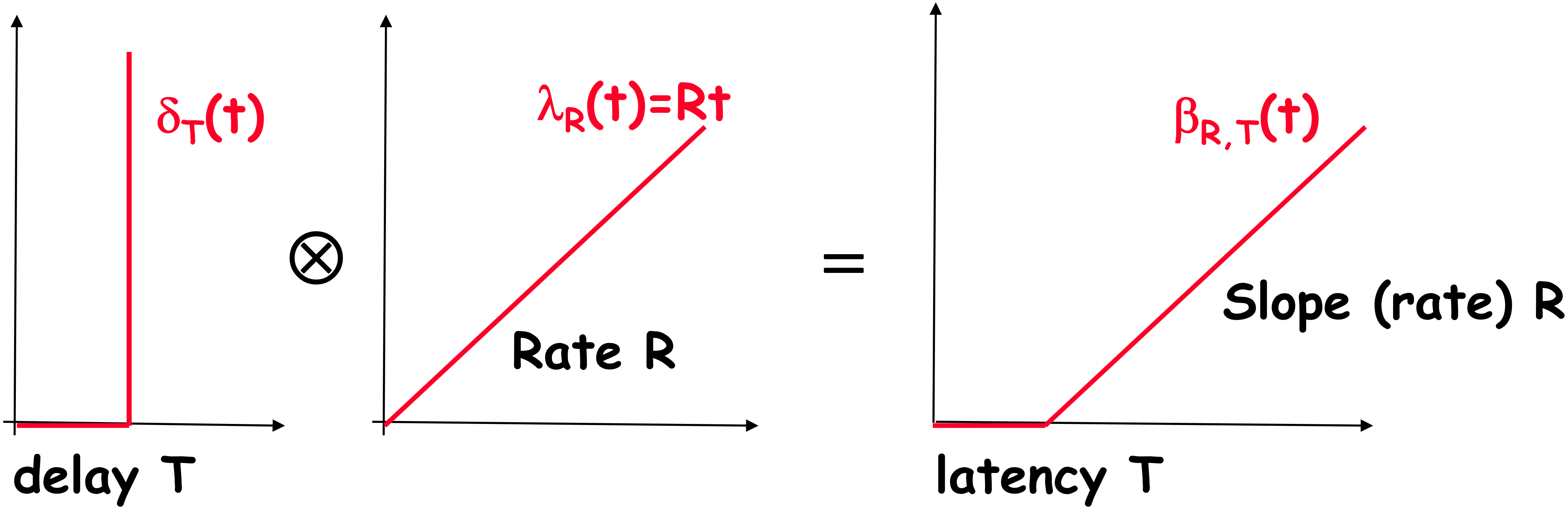
$$(f \otimes g)(t) = ?$$



Some properties of min-plus convolution

- $(f \otimes g) \in \mathcal{F}$
- $\otimes$ is associative
- $\otimes$ is commutative
- Neutral element: δ_0: $f \otimes \delta_0 = f$
  ($\delta_0(t) = 0$ for $t = 0$ and $\delta_0(t) = \infty$ for $t > 0$)
- $\otimes$ is distributive with respect to min ($\wedge$)
- $\otimes$ is isotone: $f \leq f'$ and $g \leq g' \Rightarrow f \otimes g \leq f' \otimes g'$
- Functions passing through the origin (f(0) = g(0) = 0):
  $f \otimes g \leq f \wedge g$
- Concave functions passing through the origin:
  $f \otimes g = f \wedge g$
- Convex piecewise linear functions: $f \otimes g$ is the convex piecewise linear function obtained by putting end-to-end all linear pieces of f and g, sorted by increasing slopes

Example: rate latency function

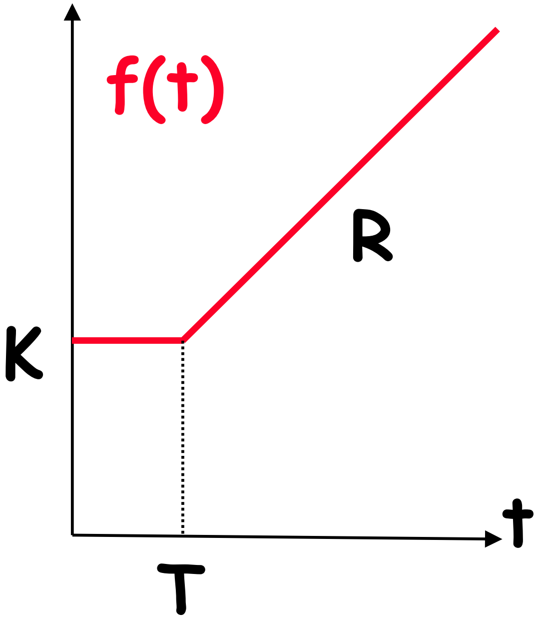


δ_T is convex
(delay function)

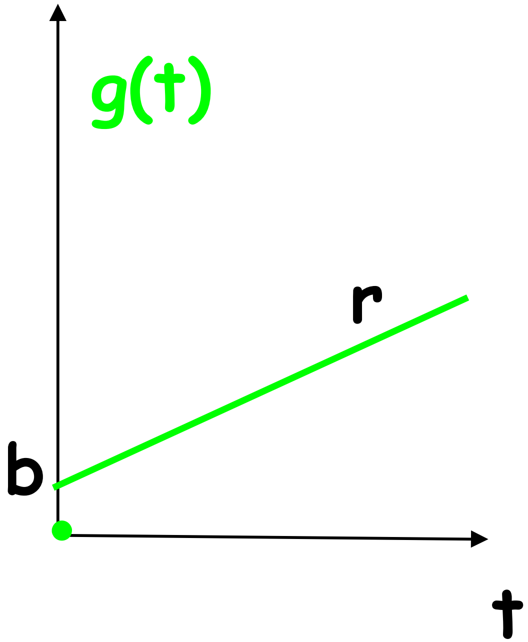
λ_R is convex
(delay function)

$\beta_{R,T}$ is convex
(rate-latency function)

Example bis (using rules)



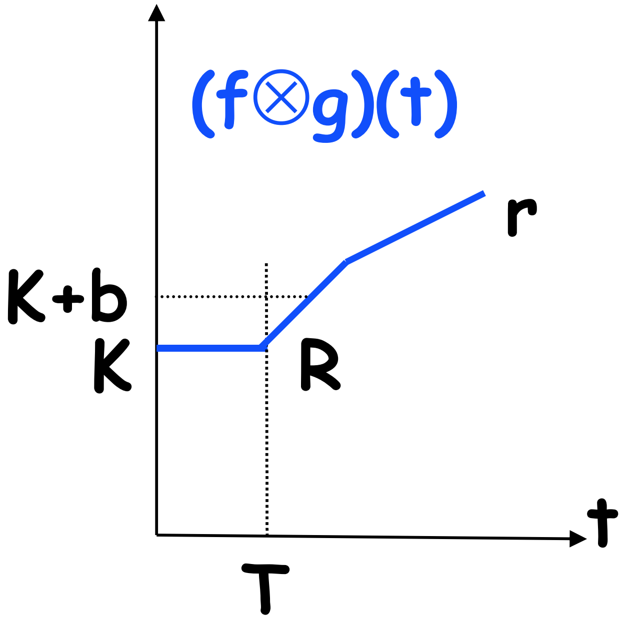
$$f = K + \beta_{R,T}$$
$$= K + \delta_T \otimes \lambda_R$$



$$g = \gamma_{r,b}$$
$$= \delta_0 \wedge (\lambda_r + b)$$

concave with $g(0) = 0$

$$
\begin{aligned}
f \otimes g &= (K + \delta_T \otimes \lambda_R) \otimes \gamma_{r,b} \\
&= K + ((\delta_T \otimes \lambda_R) \otimes \gamma_{r,b}) \\
&= K + (\delta_T \otimes (\lambda_R \otimes \gamma_{r,b})) \\
&= K + (\delta_T \otimes (\lambda_R \wedge \gamma_{r,b})) \\
&= K + (\delta_T \otimes \lambda_R) \wedge (\delta_T \otimes \gamma_{r,b}) \\
&= K + \beta_{R,T} \wedge (\delta_T \otimes \lambda_r + b) \\
&= K + \beta_{R,T} \wedge (\beta_{r,T} + b)
\end{aligned}
$$



We can express arrival curves with min-plus convolution

- Arrival Curve property means for all $0 \leq s \leq t$,

$$x(t) - x(s) \leq \alpha(t-s)$$

$\Leftrightarrow$ $x(t) \leq x(s) + \alpha(t-s)$ for all $0 \leq s \leq t$

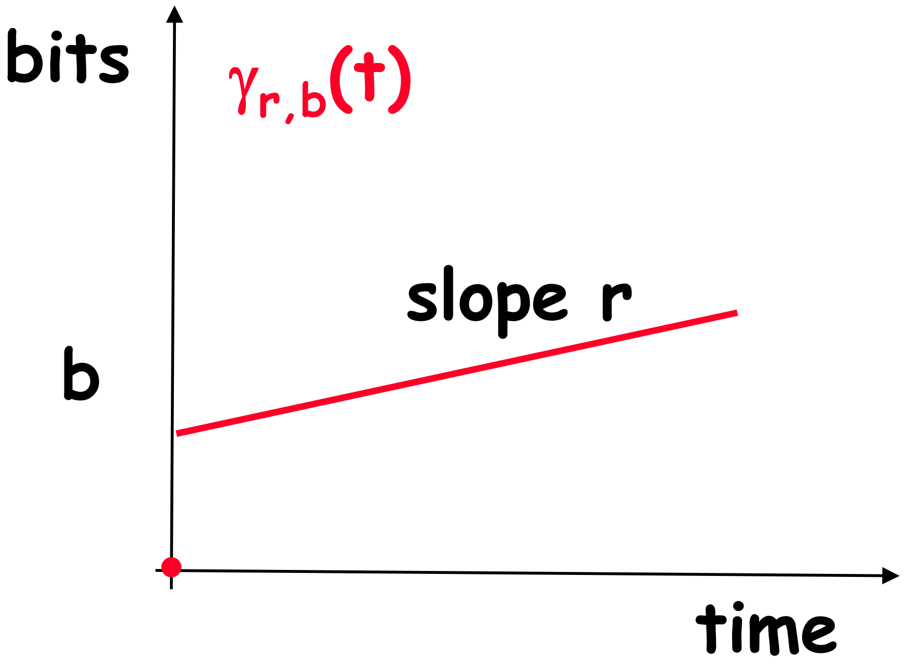
$\Leftrightarrow$ $x(t) \leq \inf_u \{ x(u) + \alpha(t-u) \}$

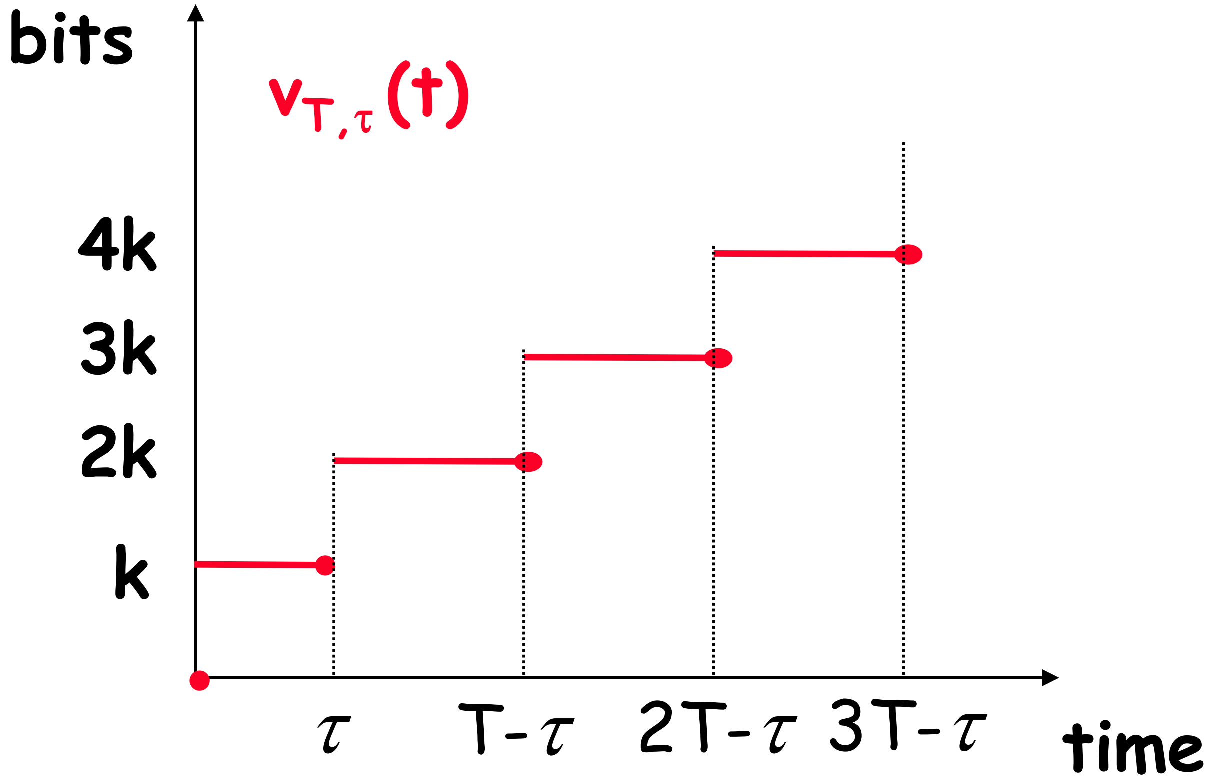
$\Leftrightarrow$ $x \leq x \otimes \alpha$

Sub-additive function

- f is sub-additive $\Leftrightarrow$ $f(t) + f(s) \geq f(t+s)$
- f is concave with f(0) = 0 $\Rightarrow$ f is sub-additive
- f is sub-additive $\not\Rightarrow$ f is concave
- f,g are sub-additive and pass through the origin (f(0) = g(0) = 0) $\Rightarrow$ $f \otimes g$ is sub-additive

Examples





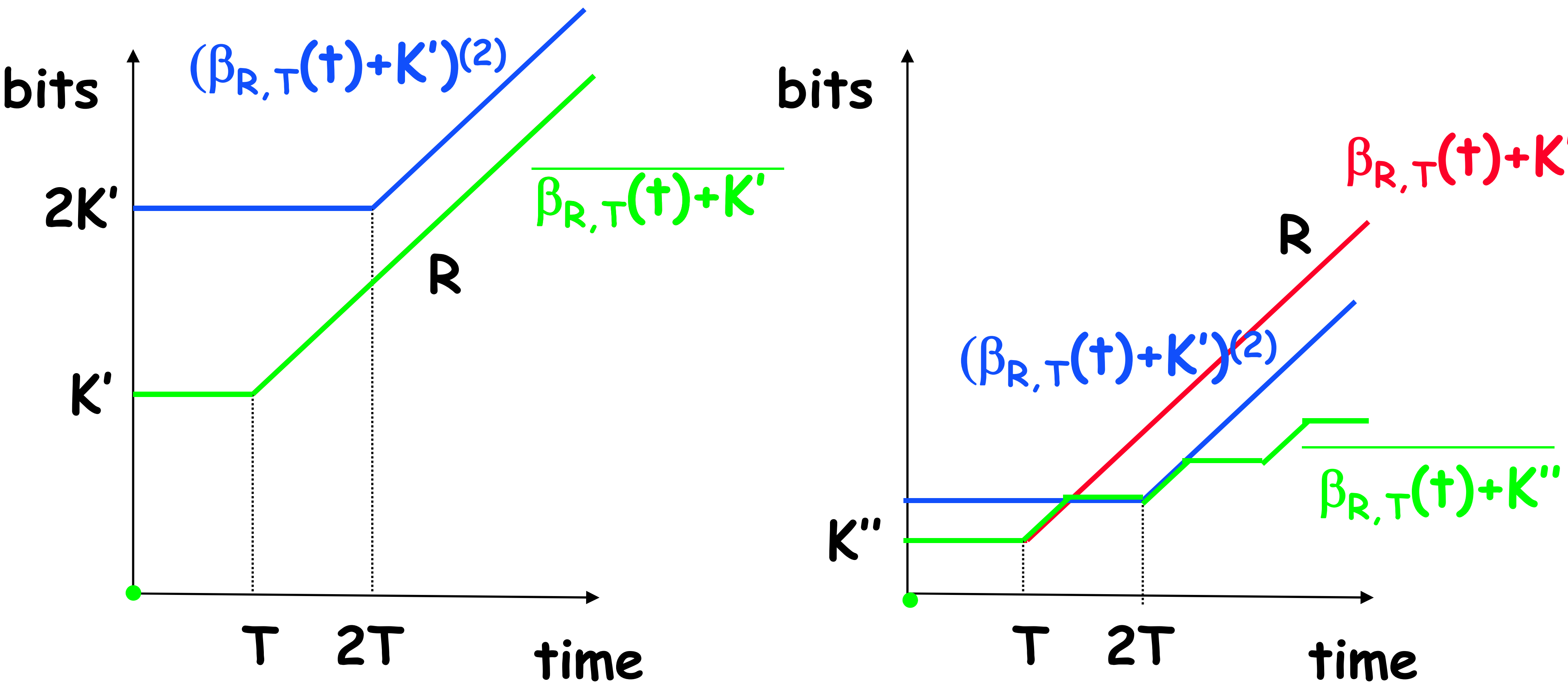
$\gamma_{r,b}$ is concave

$v_{T,\tau}$ is not concave, but is sub-additive

Sub-additive closure

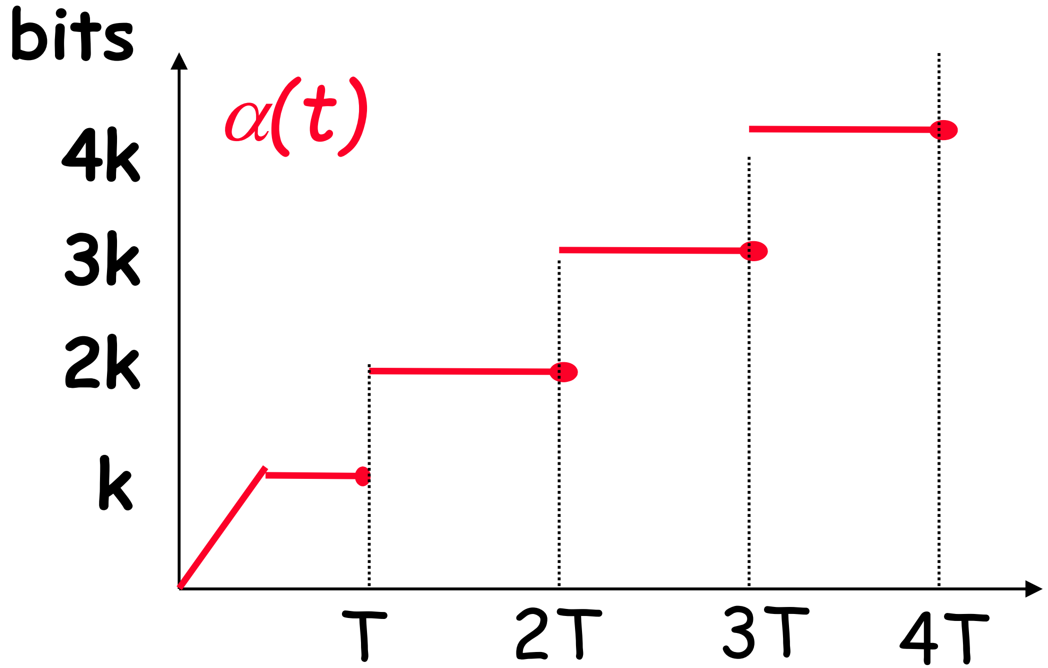
- $\overline{f} = \inf \{\delta_0, f, f \otimes f, f \otimes f \otimes f, \ldots\}$
- $\overline{f}$ is sub-additive with f(0) = 0
- f is sub-additive with f(0) = 0 $\Leftrightarrow \overline{f} = f \Leftrightarrow f = f \otimes f$

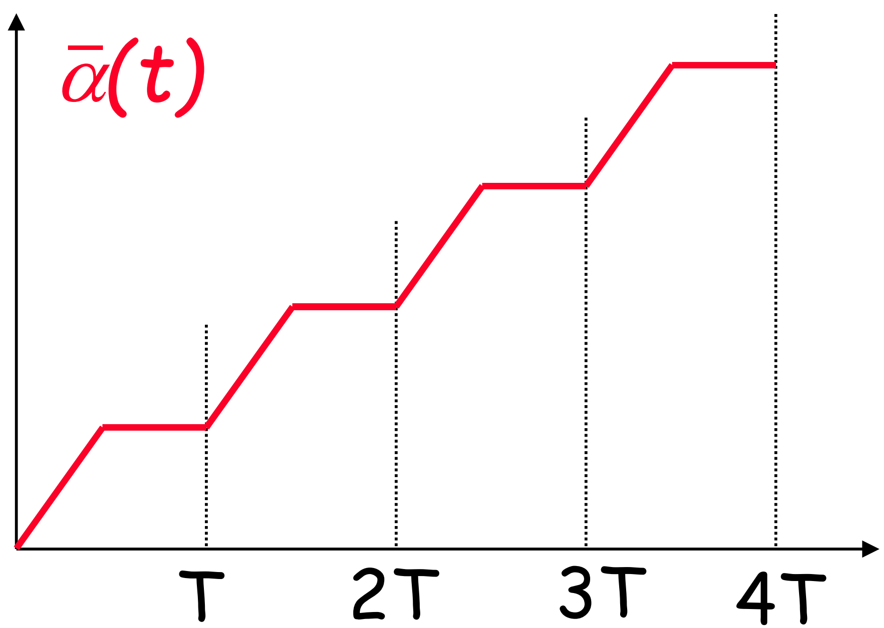
Examples


bits
$(\beta_{R,T}(t)+K')^{(2)}$
2K'
$\overline{\beta_{R,T}(t)+K'}$
R
K'
T
2T
time
bits
$\beta_{R,T}(t)+K''$
R
$(\beta_{R,T}(t)+K')^{(2)}$
$\overline{\beta_{R,T}(t)+K''}$
K''
T
2T
time

Sub-additivity and arrival curves

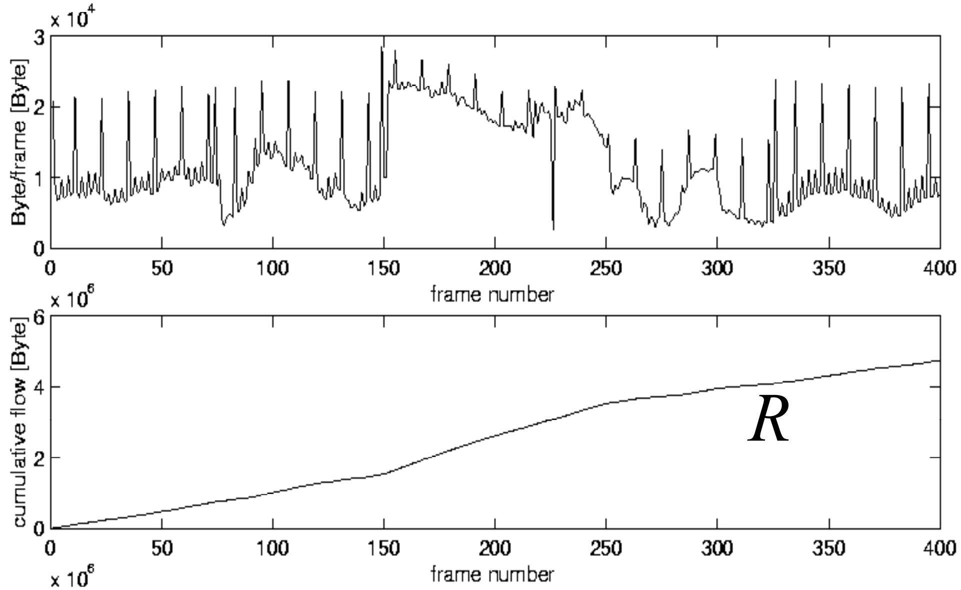
- What is $\bar{\alpha}(t)$?
- α can be replaced by its sub-additive closure $\bar{\alpha}$.
- From now on: we will always take sub-additive arrival curves passing through the origin.





Minimal arrival curve

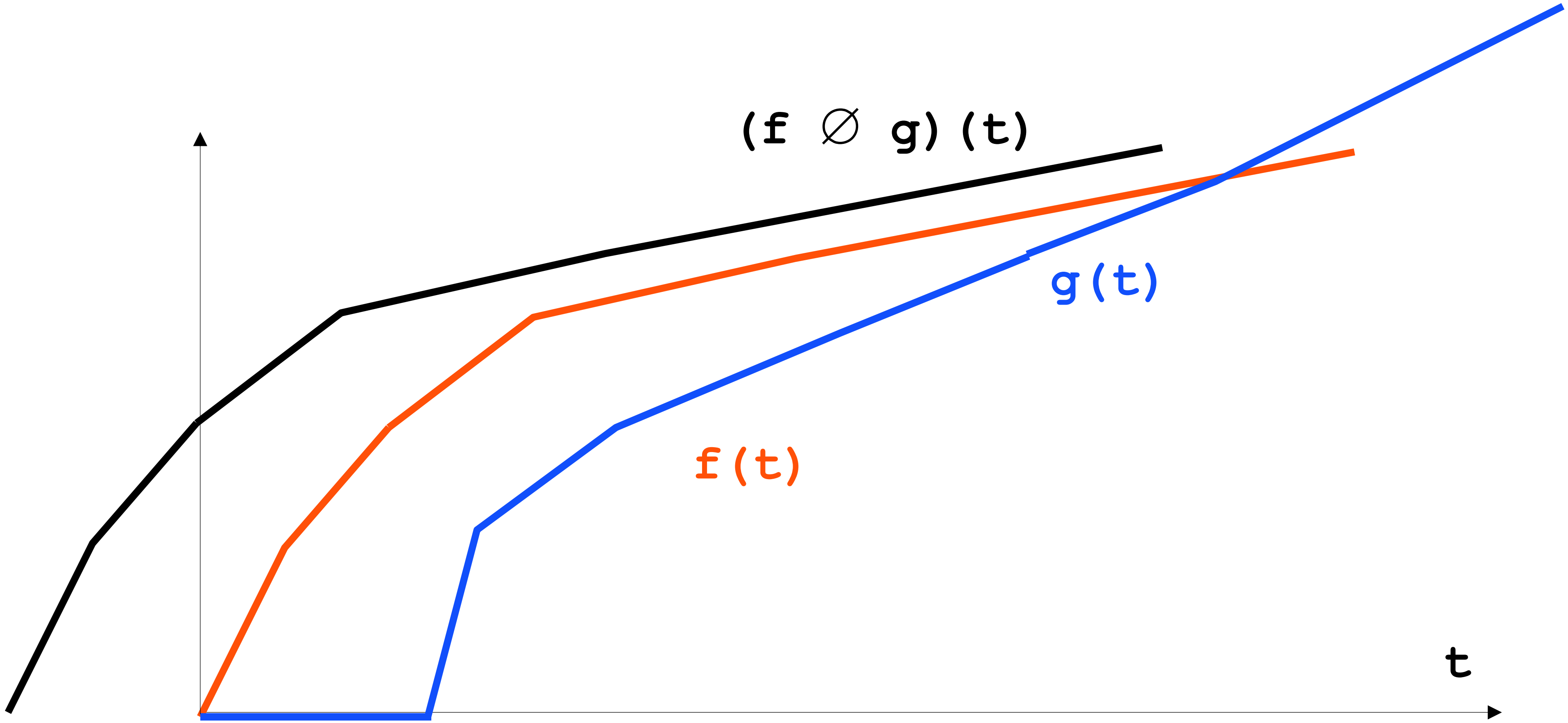
- If the only available information on a flow is obtained from measurements, i.e if we only know R, how can we compute its minimal arrival curve α ?
- The answer uses min-plus deconvolution



Min-plus deconvolution ⌀

- Definition

$$(f \oslash g)(t) = \sup_u \{ f(t+u) - g(u) \}$$



Some properties of min-plus deconvolution

- $(f \oslash g) \notin \mathcal{F}$ in general
- $(f \oslash f) \in \mathcal{F}$
- $(f \oslash f)$ is sub-additive with $(f \oslash f)(0) = 0$
- $(f \oslash g) \oslash h = f \oslash (g \otimes h)$
- Duality with $\otimes$: $f \oslash g \leq h \Leftrightarrow f \leq g \otimes h$

Minimal arrival curve

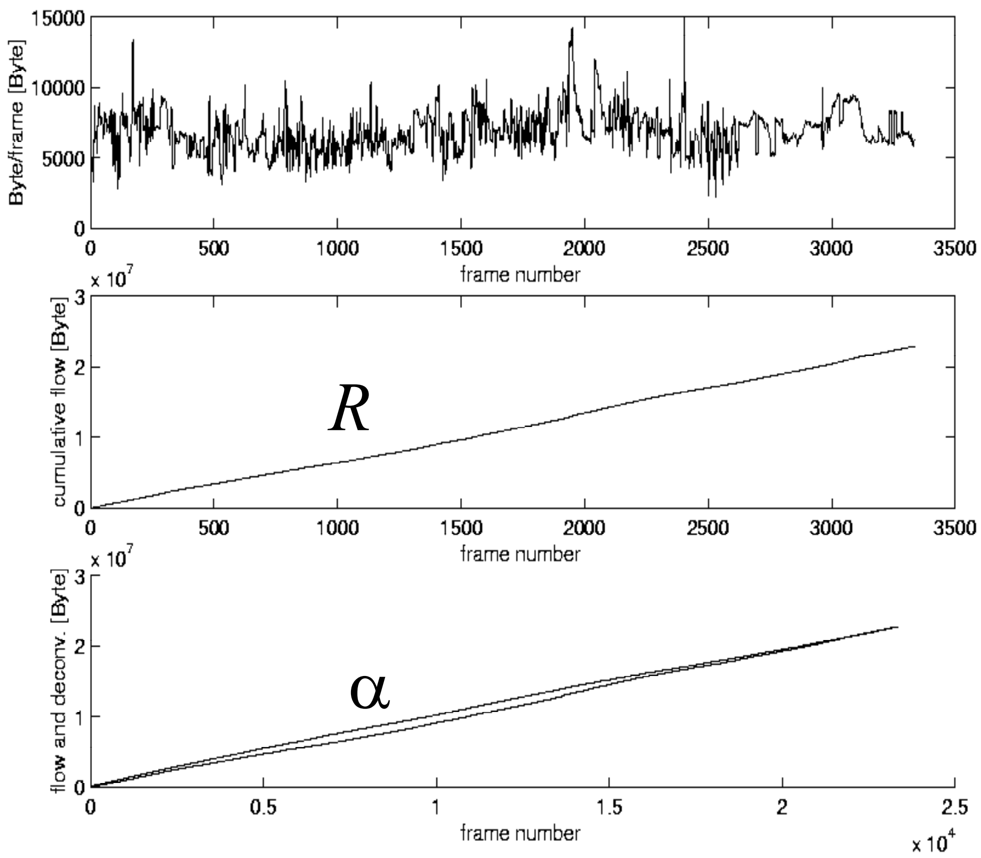
- The minimal arrival curve of flow R is $\alpha = R \oslash R$.
- Proof:
  - It is an arrival curve because

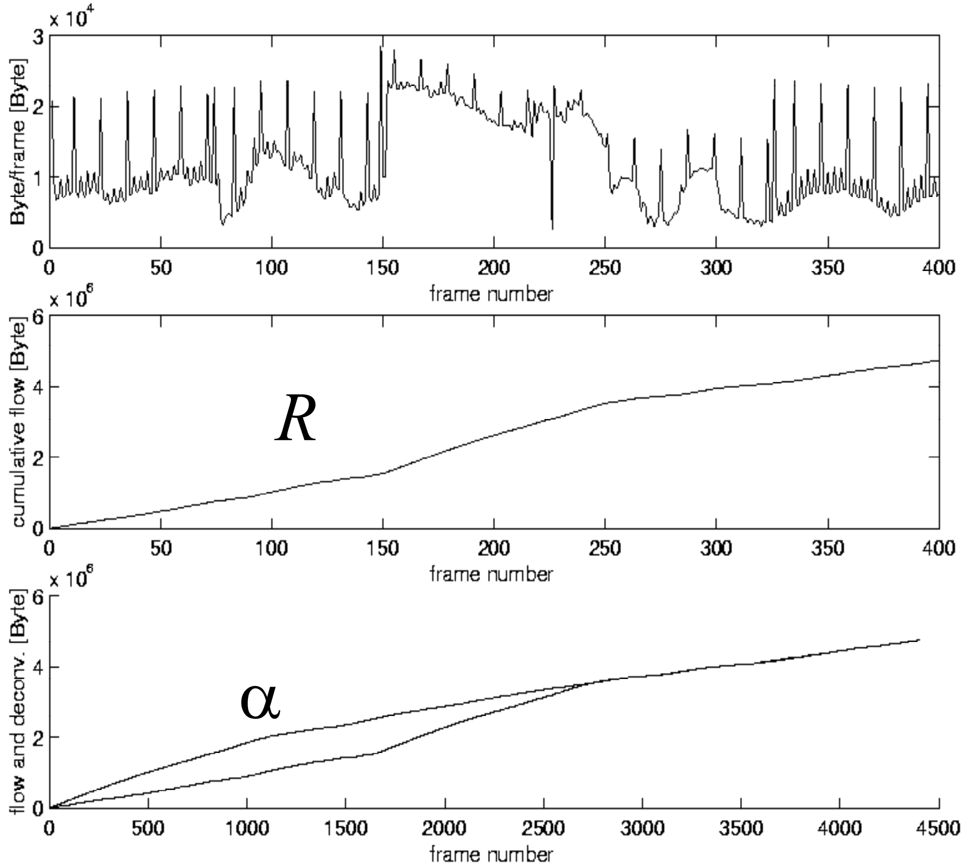
$$R(t) - R(s) = R((t-s)+s) - R(s)$$
$$\leq \sup_u \{ R((t-s)+u) - R(u) \} = (R \oslash R)(t-s)$$

  - If α' is another arrival curve for flow R, then $R \leq R \otimes \alpha'$
    $\Leftrightarrow R \oslash R \leq \alpha'$ so that $\alpha \leq \alpha'$.

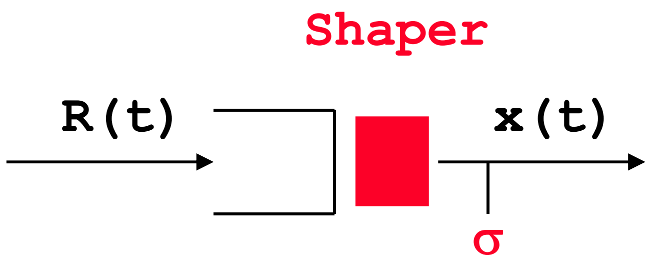
Example

- MPEG files, 25 frames/sec





Greedy shaper



Definition of Greedy shaper

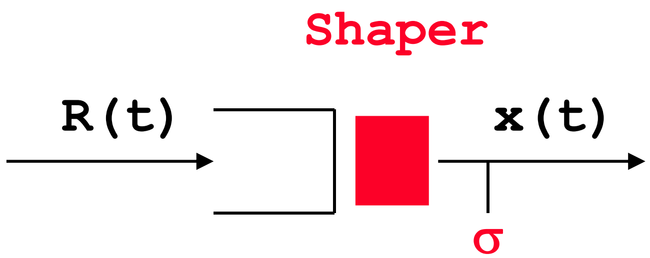
- forces output to be constrained by arrival curve σ

$$x(t) - x(s) \leq \sigma(t - s)$$

- stores data in a buffer if needed
- Hence the shaper maximises $x(t)$ such that

$$x(t) \leq R(t)$$
$$x(t) \leq (x \otimes \sigma)(t)$$

Output of a Greedy shaper



- If σ is sub-additive and $\sigma(0) = 0$, $x = R \otimes \sigma$
- Proof:
  - $x = R \otimes \sigma$ is a solution because

    $x = R \otimes \sigma \leq R$ since $\sigma(0) = 0$

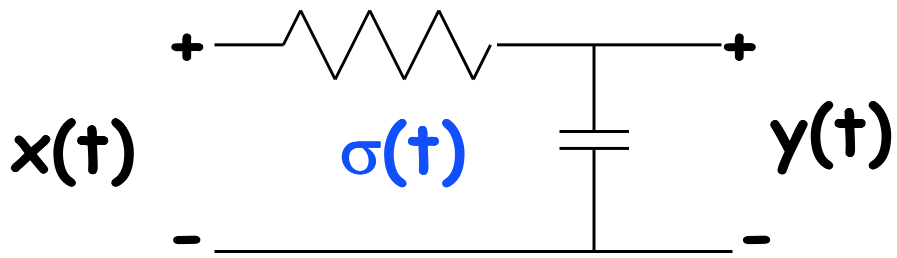
    $x = R \otimes \sigma = R \otimes (\sigma \otimes \sigma) = (R \otimes \sigma) \otimes \sigma = x \otimes \sigma$
  - If x' is another solution then $x' \leq R$ and $x' \leq x' \otimes \sigma$. Combining the two and using isotonicity of $\otimes$:

$$x' \leq x' \otimes \sigma \leq R \otimes \sigma = x$$

Greedy shaper = linear min-plus filter

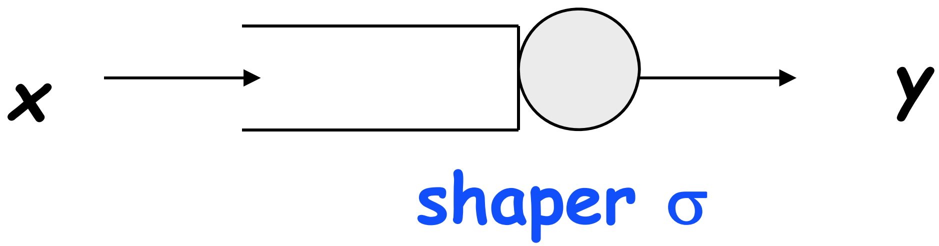
- Standard convolution in (R, x, +) (LTI filter)

$$y(t) = (\sigma * x)(t) = \int \sigma(t-u)\, x(u)\, du$$

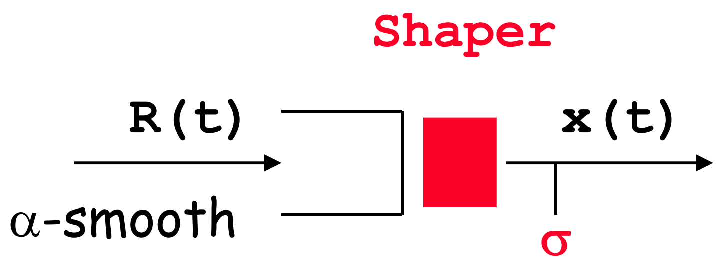


- Min-plus convolution in (R, +, $\wedge$) is linear ($\wedge$ = min)

$$y(t) = (\sigma \otimes x)(t) = \inf_u \{ \sigma(t-u) + x(u) \}$$



What is done by shaping cannot be undone by shaping

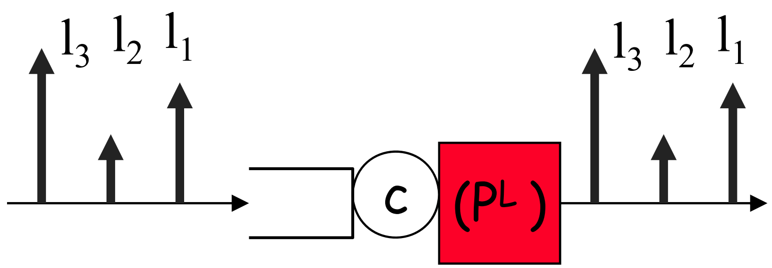


- Suppose that R(t) is constrained by arrival curve $\alpha : R \leq R \otimes \alpha$.
- Then $x = R \otimes \sigma \leq (R \otimes \alpha) \otimes \sigma = R \otimes (\alpha \otimes \sigma) \leq R \otimes \alpha$ since $\sigma(0) = 0$.
- Therefore shaping keeps arrival constraints.
- In fact, the output flow has $\alpha \otimes \sigma$ as arrival curve

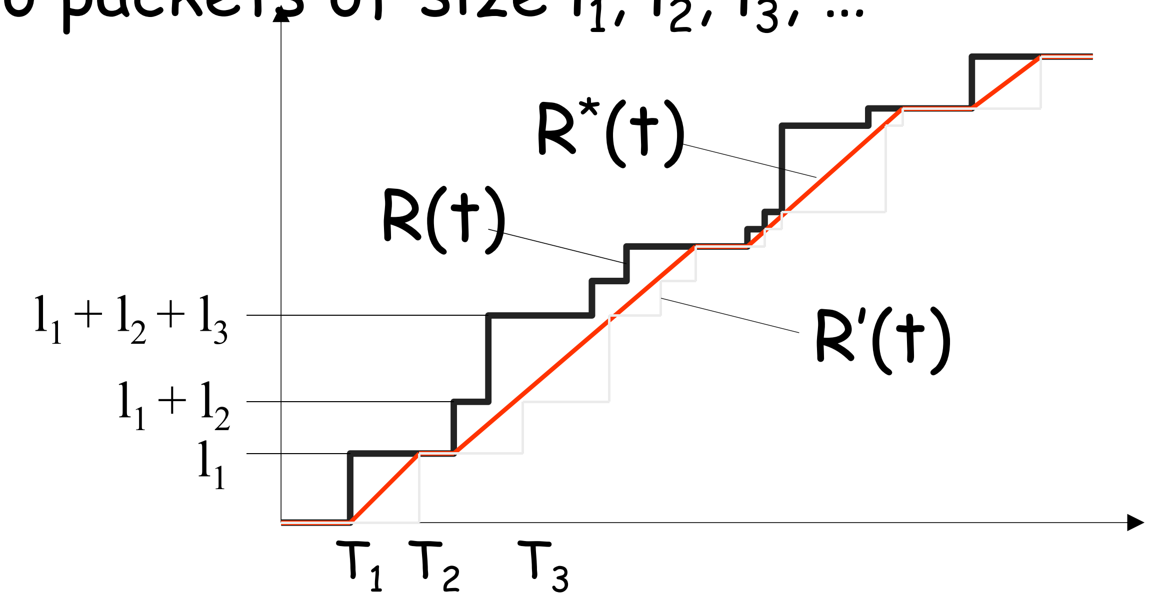
Packetization

- The shaper presented before is for constant size packets or ideal fluid systems
- Real life systems are modelled by adding a packetizer

  transforms fluid input into packets of size l_1, l_2, l_3, ...



constant rate server
=
greedy shaper $\sigma(t)$=ct
+ packetizer



- Packetizer adds some distortion, well understood

Contents

1. Arrival curves

2. Service curves, backlog, delay bounds

- Service curve: definition
- Backlog and delay bounds
- Guaranteed Rate Servers
- Application to IntServ
- Application to Core-Stateless

3. Diffserv: intuition and formal definition behind EF

4. Min-plus algebra in action: Video smoothing

5. Statistical multiplexing with EF

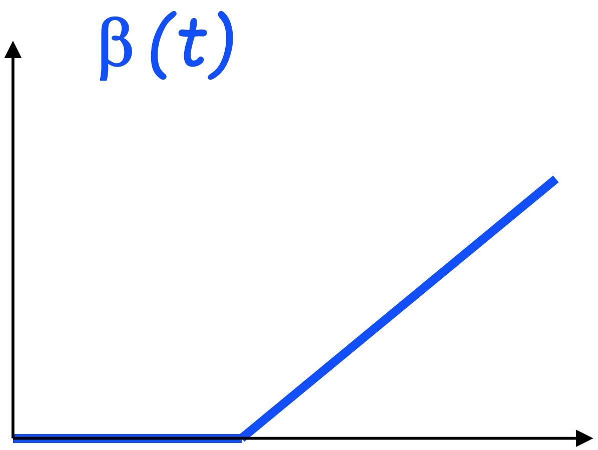
Goal of Service Curve and GR node definitions

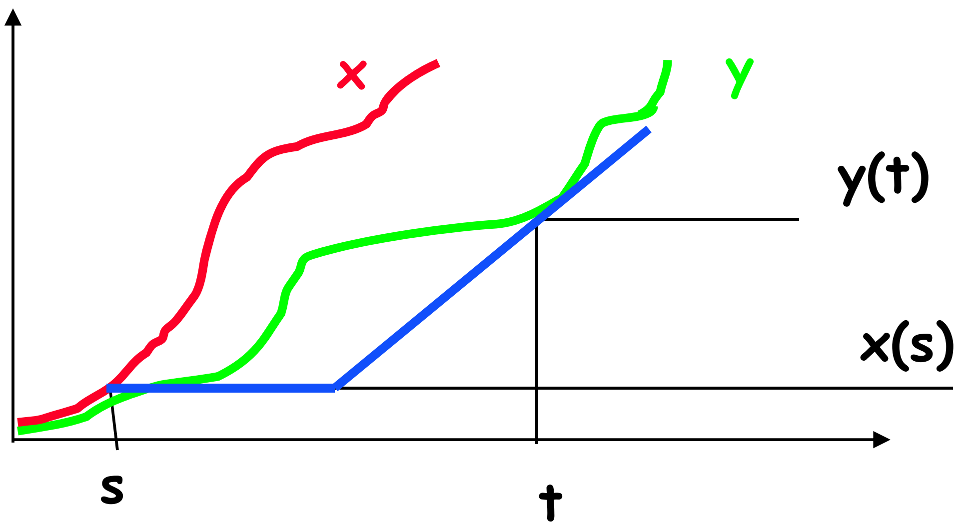
- define an abstract node model
- independent of a specific type of scheduler
- applies to real routers, which are not a single scheduler, but a complex interconnection of delay and scheduling elements
- applies to nodes that are not work-conserving

Service Curve

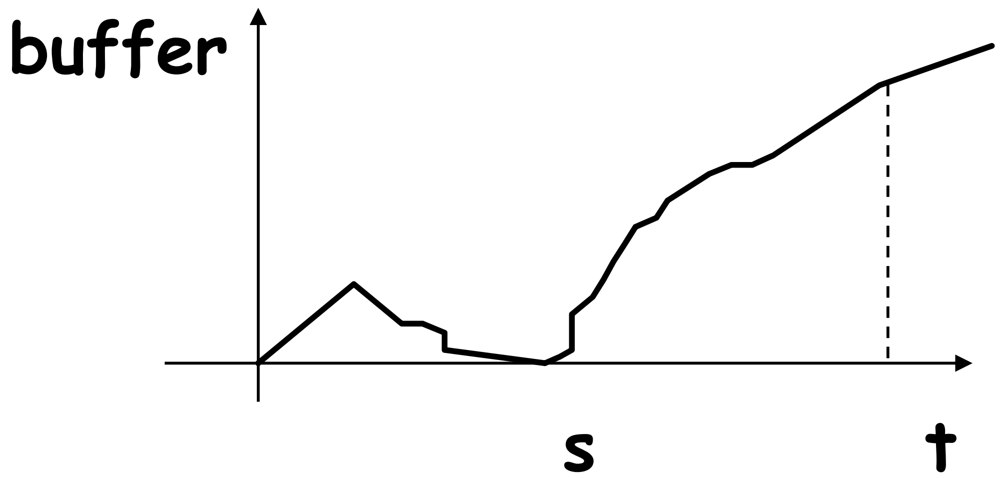
- System S offers a (minimal) service curve β to a flow iff for all t there exists some s such that

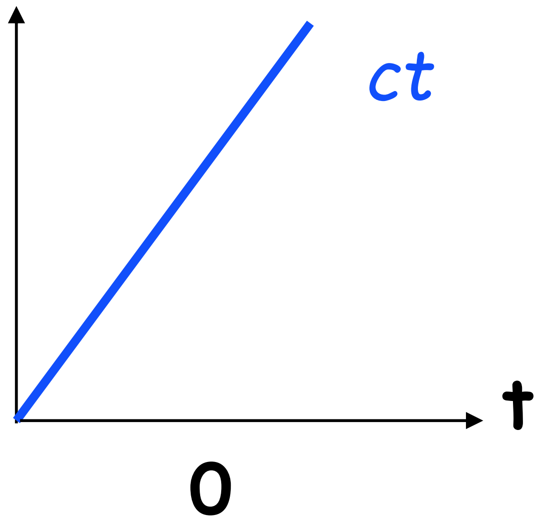
$$y(t) - x(s) \geq \beta(t-s)$$





The constant rate server has service curve $\beta(t)=ct$



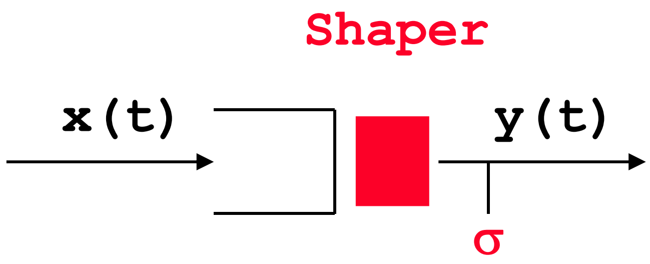


Proof: take s = beginning of busy period:

$$y(t) - y(s) = c\,(t-s) \quad \text{and} \quad y(s) = x(s)$$

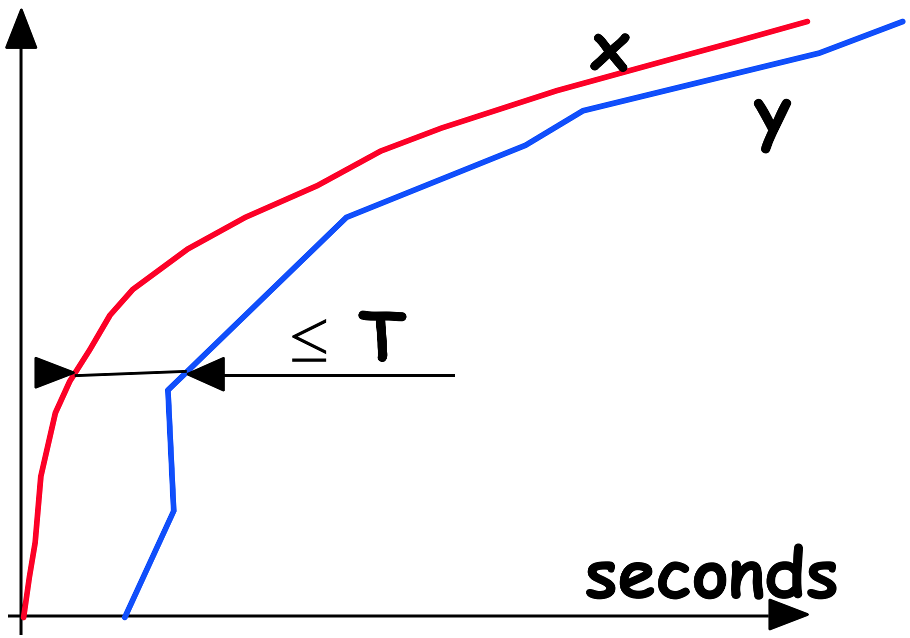
$$\to \; y(t) - x(s) = c\,(t-s)$$

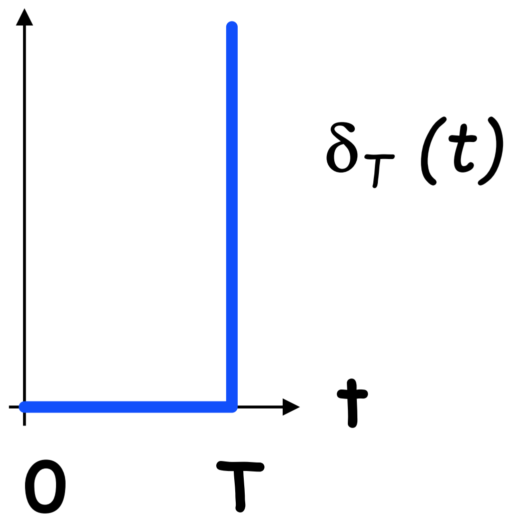
The service curve of a Greedy shaper is its shaping curve



- If σ is sub-additive and $\sigma(0) = 0$, $y(t) = (x \otimes \sigma)(t)$.
- The service curve of a shaper is thus σ.

The guaranteed-delay node has service curve δ_T

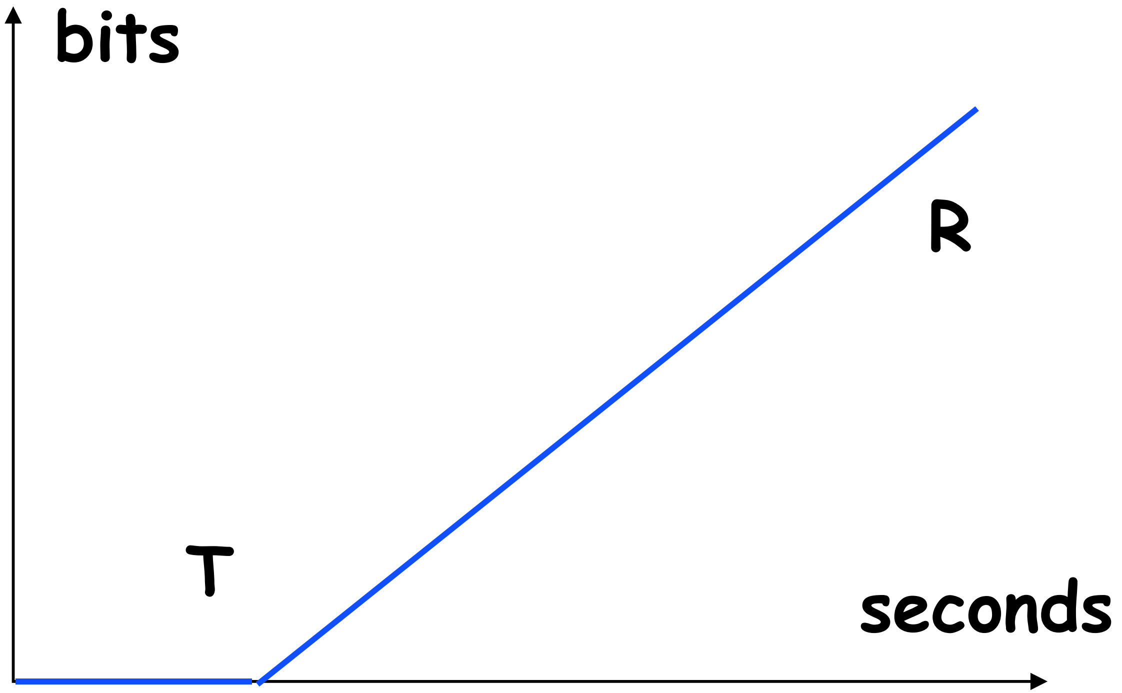




Function δ_T

The standard model for an Internet router

- rate-latency service curve



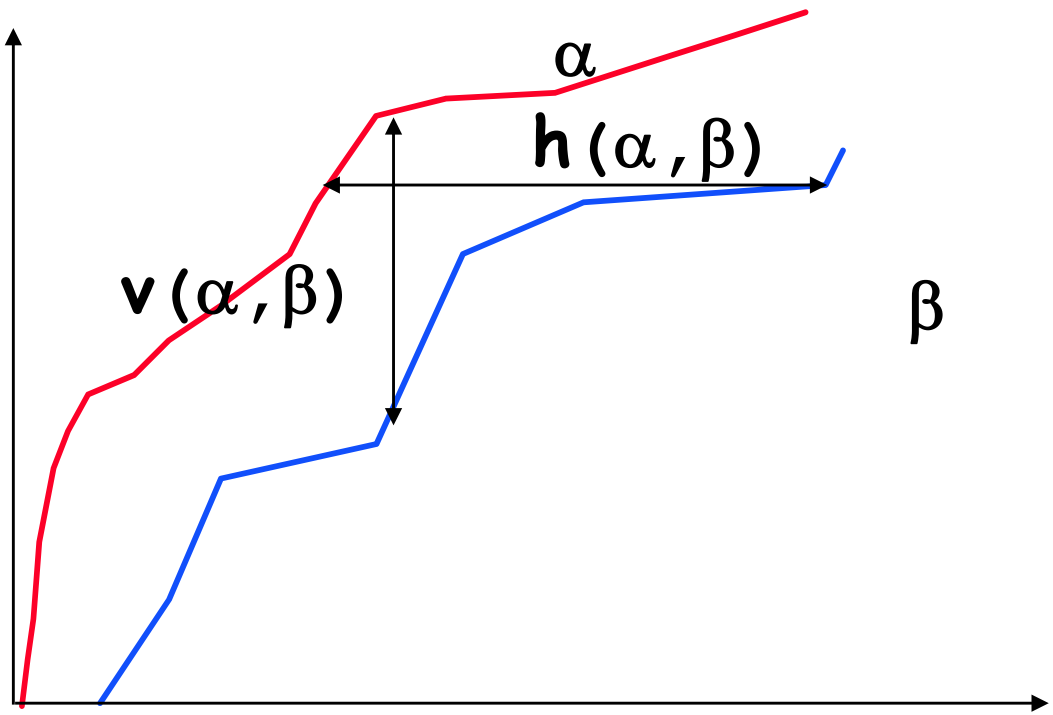
We can express service curves with min-plus convolution

- Service Curve guarantee means there exists some $0 \leq s \leq t$: $y(t) - x(s) \geq \beta(t-s)$
  - $\Leftrightarrow$ $y(t) \geq x(s) + \beta(t-s)$ for some $0 \leq s \leq t$
  - $\Leftrightarrow$ $y(t) \geq \inf_u \{ x(u) + \beta(t-u) \}$
  - $\Leftrightarrow$ $y \geq x \otimes \beta$

Tight Bounds on delay and backlog

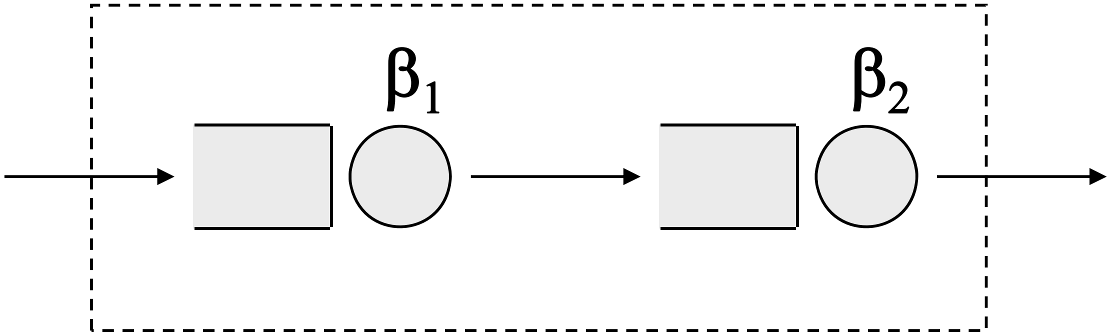
If flow has arrival curve α and node offers service curve β then

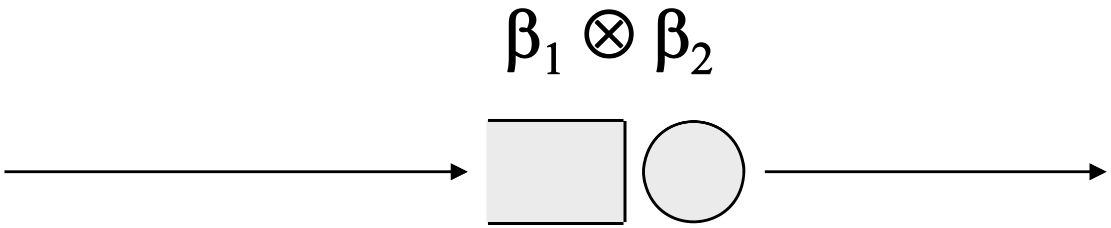
- backlog $\leq \sup (\alpha(s) - \beta(s)) = (\alpha \oslash \beta)(0) = v(\alpha, \beta)$
- delay $\leq \inf \{ s \geq 0 : (\alpha \oslash \beta)(-s) \leq 0 \} = h(\alpha, \beta)$



The composition theorem

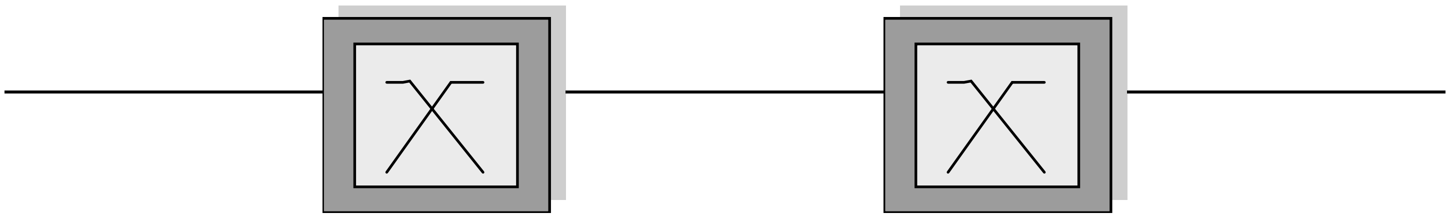
- **Theorem**: the concatenation of two network elements each offering service curve β_i offers the service curve $\beta_1 \otimes \beta_2$

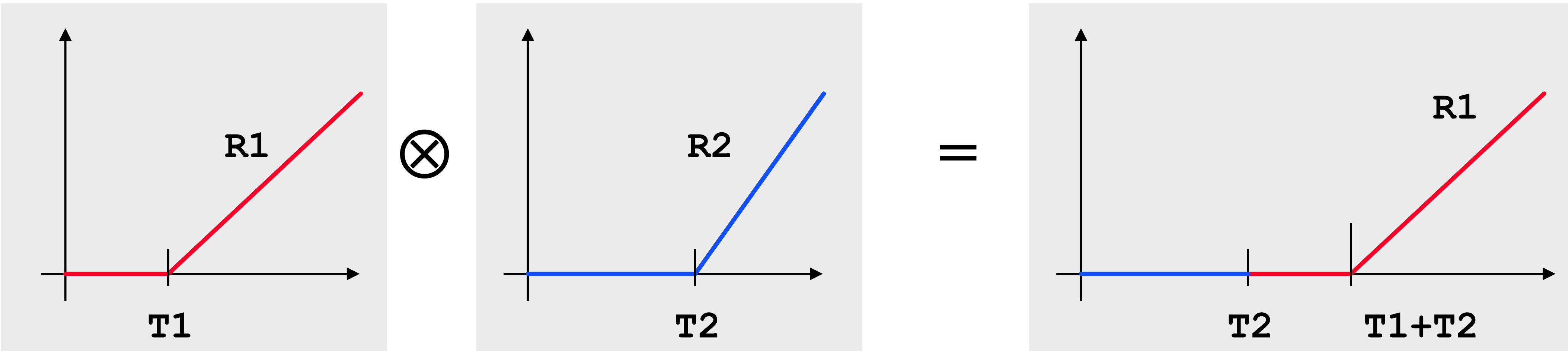




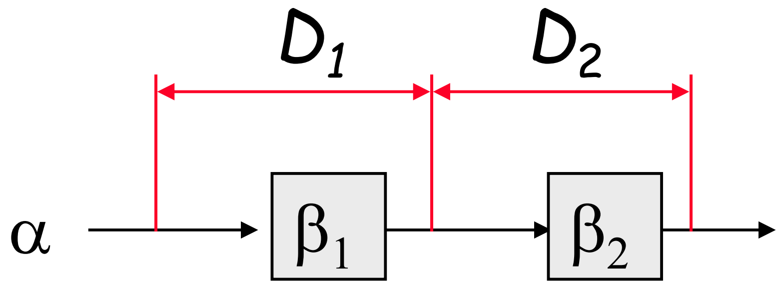
Example

- tandem of routers

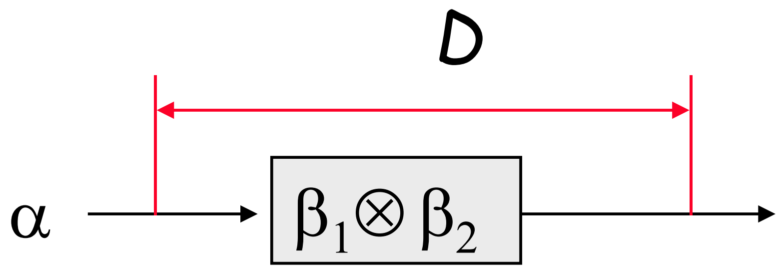




Pay Bursts Only Once



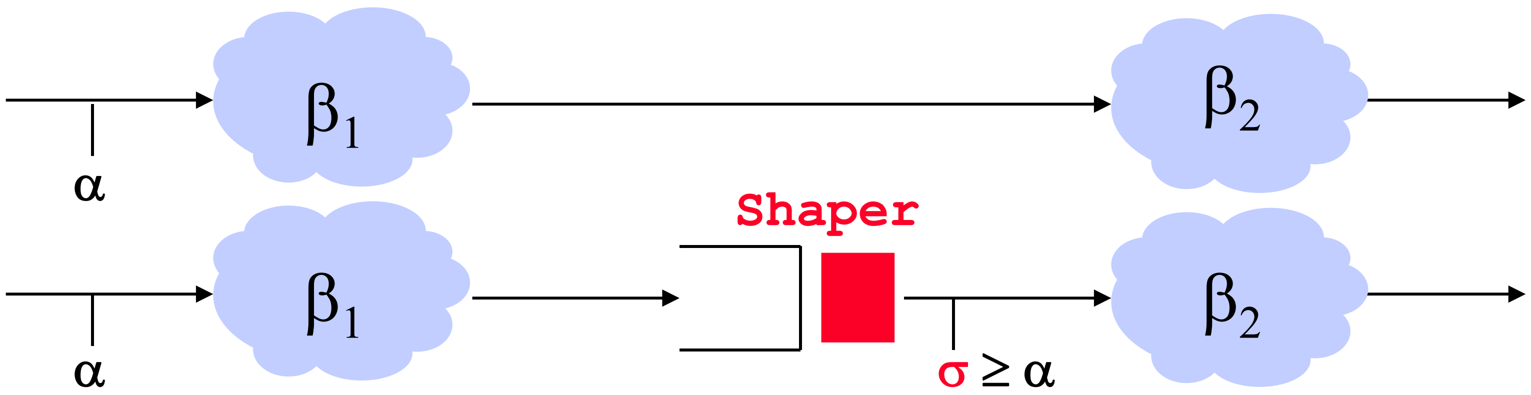
$$D_1 + D_2 \leq (2b + RT_1)/R + T_1 + T_2$$



$$D \leq b/R + T_1 + T_2$$

end to end delay bound is less

Re-shaping is for free



end-to-end service curve $\beta_1 \otimes \beta_2$

$\beta_1 \otimes \sigma \otimes \beta_2$

- Re-shaper is added to re-enforce some fraction of the original constraint
- Delay for original system = h(α, $\beta_1 \otimes \beta_2$)
- For system with re-shaper = h(α, $\beta_1 \otimes \sigma \otimes \beta_2$) = h($\alpha$, $\sigma \otimes \beta_1 \otimes \beta_2$)
- Now h(α, $\sigma \otimes \beta_1 \otimes \beta_2$) = h($\alpha$, $\beta_1 \otimes \beta_2$)

  interpretation: put re-shaper before node 1; it is transparent

  formal proof uses delay = inf { d : $\alpha \oslash (\beta_1 \otimes \beta_2)(-d) \le 0$ }

- Therefore delay bound for both systems are equal

Guaranteed Rate node

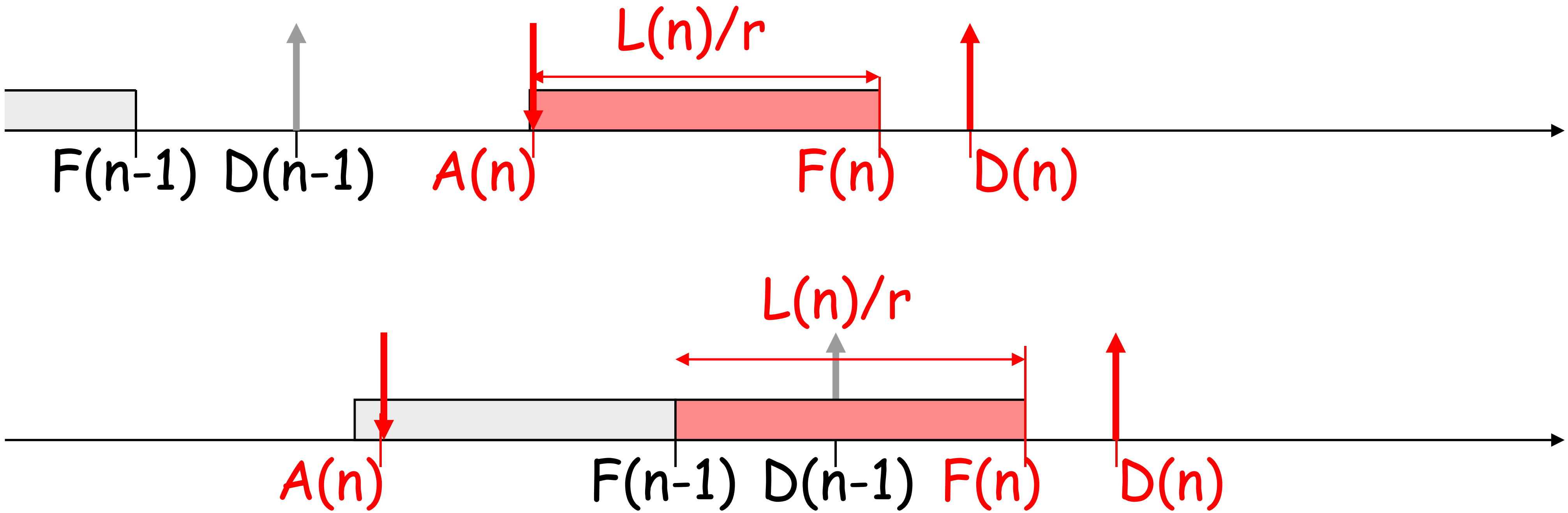
- An alternative definition to service curve for FIFO
  - for rate-latency service curves only
- Definition (Goyal, Lam, Vin; Chang): a node is GR(r,e) if

$$D(n) \leq F(n) + e$$

$$F(n) = \max\{A(n), F(n-1)\} + L(n)/r$$

D(n) : departure time for packet n
A(n) : arrival time
F(n) : virtual finish time, F(0) = 0
L(n) : length in bits for packet n

$$F(n) = \max\{A(n), F(n-1)\} + L(n)/r$$



GR is equivalent to rate-latency service curve -- for FIFO per flow

- GR(r,e) is equivalent to

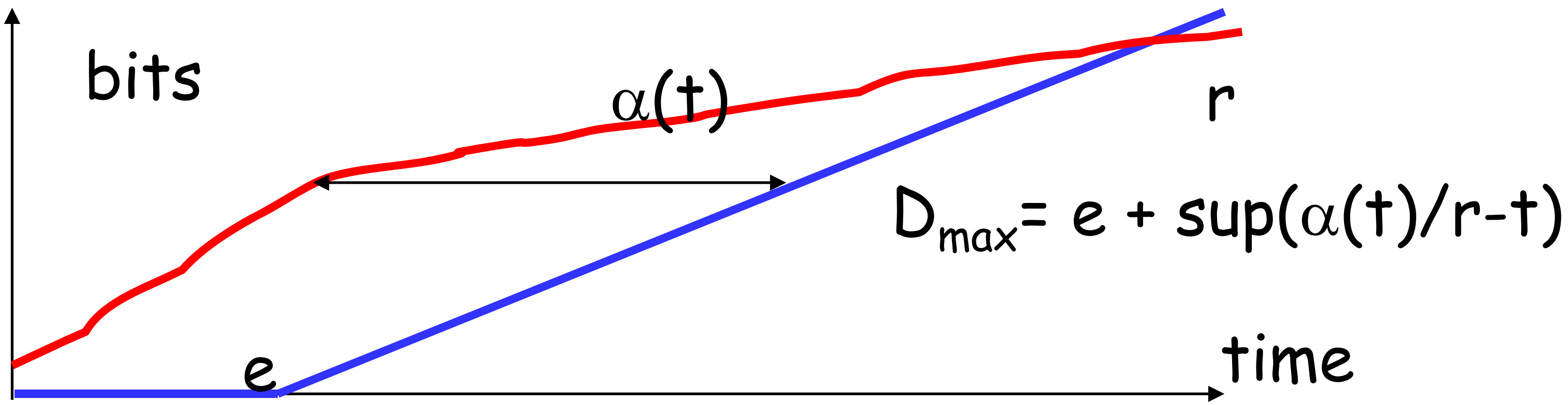
$$D(n) \leq \max_{k \leq n}[A(k) + (L(k) + \ldots + L(n))/r] + e$$

  - max-plus analog to service curve
- **Theorem (equivalence for FIFO per flow nodes):**
  - a GR node is a service curve element with rate-latency service curve (r,e) followed by a packetizer
  - conversely, consider a node which is FIFO per flow and serves entire packets. If it has the rate-latency service curve (R,T) then it is GR(R,T).
- FIFO per flow is true in IntServ context

Properties of GR nodes (FIFO per flow or not)

- delay bound = $h(\alpha, \beta)$

$$D_{max} = e + \sup[\alpha(t)/r - t]$$



for FIFO per flow nodes = delay at service curve element (packetizer does not add per-packet delay)

- backlog bound = $v(\alpha, \beta) + L_{max}$

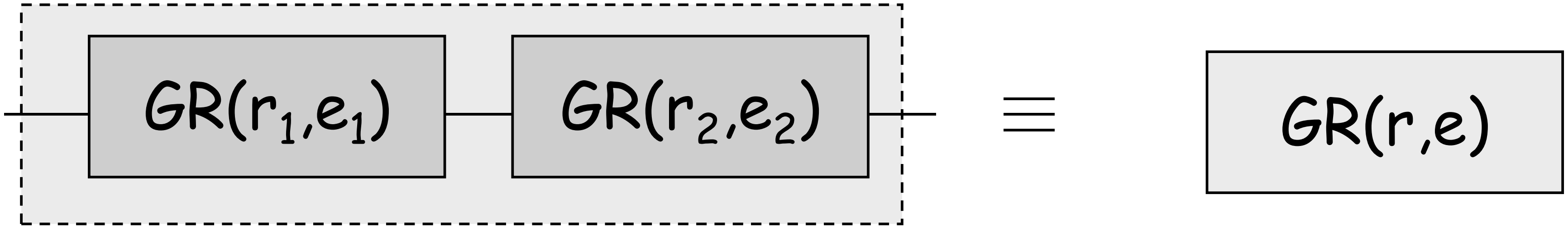
$$B_{max} = \sup[\alpha(t) - R(t-T)^{+}] + L_{max}$$

Modelling a node with GR

- queue with rate C: R=C, T=0
- priority queue with rate c: R=C, $T=L_{max}/C$
- element with bounded delay d: $R = \infty$, T=d
- and combine these elements

Concatenation of GR nodes

- FIFO per flow nodes: apply service curve rule

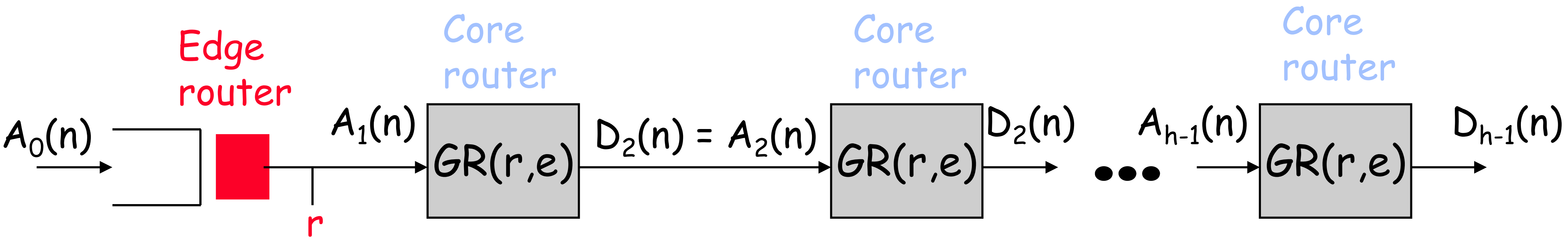


$$r = \min(r_1, r_2)$$
$$e = e_1 + e_2 + L_{max}/r_1$$

- non FIFO per-flow: not true (LeBoudec Charny, Infocom 2002)

Core-Stateless

- Imagine routers can maintain flow state information and offer per flow guarantees



q rate-based routers GR(0,r): $A_i(n) = max(A_{i-1}(n), A_i(n-1)) + L(n)/r$

h-q delay-based routers GR(e,∞): $A_i(n) = max(A_{i-1}(n), A_i(n-1)) + e$

Arrival curve at core edge is $\alpha(t) = \gamma_{r,Lmax}(t) = L_{max} + rt$

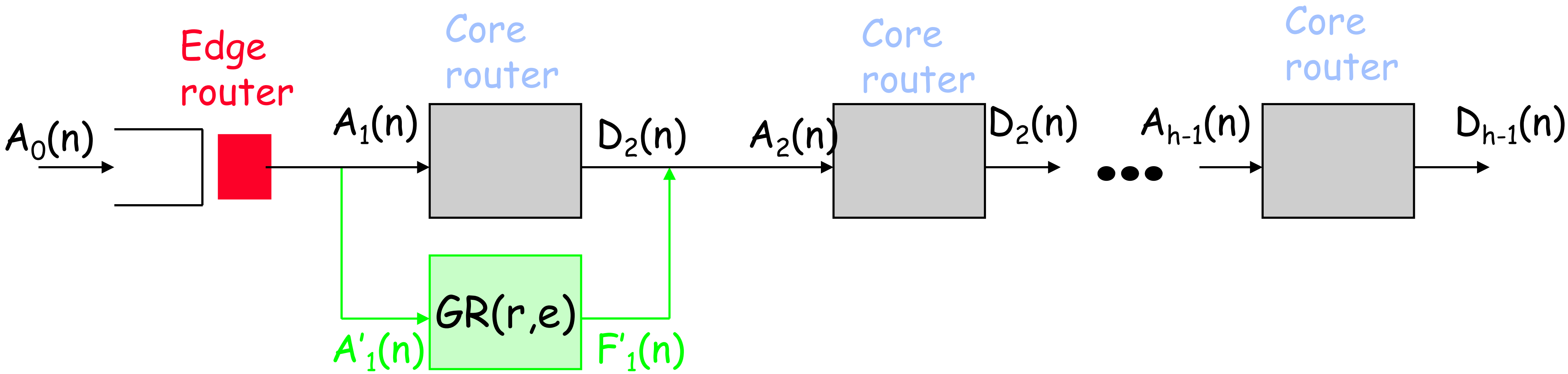
Service curve for the core is $\beta(t) = \beta_{r,(h-q)e+(q-1)Lmax/r}(t)$

$$= r[t-((h-q)e + (q-1)L_{max}/r)]^+$$

-> Delay bound is $h(\alpha, \beta) = (h-q)e + qL_{max}/r$
(neglecting propagation delays)

Core-Stateless

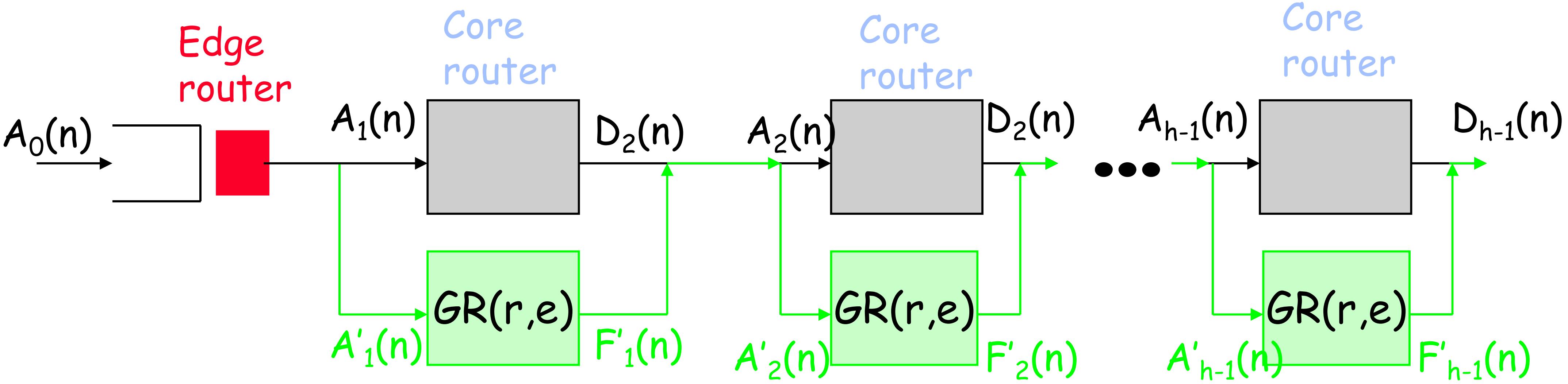
- Core routers do not maintain per flow information



- Replace it by the use of timestamps, with the virtual arrival time: virtual arrival time at router *i* for *n*th packet is $A'_i(n)$
- NB: A virtual delay adjustment is also inserted (not considered here) (it is the delay that *n*th packet would have experienced in the ideal system with flow state information – $qL(n)/r$)
- Compute the virtual finish time $F'_i(n) = A'_i(n) + L(n)/r + e$

Core-Stateless

- Core routers do not maintain per flow information

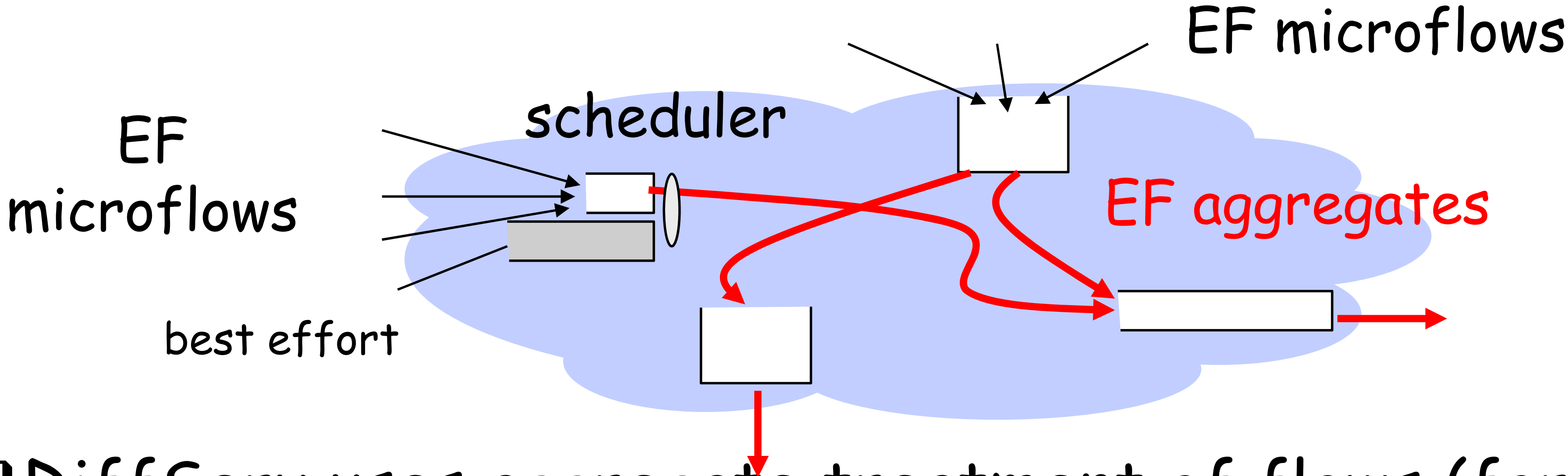


- The actual departure time $D_i(n)$ of the nth packet from the *i*th core router is less than $F'_i(n)$ + processing delay
- Update virtual arrival time: $A'_{i+1}(n) = F'_i(n)$ + processing delay + propagation delay
- Property: with q rate-based routers and h-q delay-based routers, delay bound is $h(\alpha, \beta) = (h-q)e + qL_{max}/r$ + sum of processing delays + sum of propagation delays

Contents

1. Arrival curves
2. Service curves, backlog, delay bounds
3. Diffserv: intuition and formal definition behind EF
   - PSRG
   - non-FIFO and min-max
   - SETF
4. Min-plus algebra in action: Video smoothing
5. Statistical multiplexing with EF

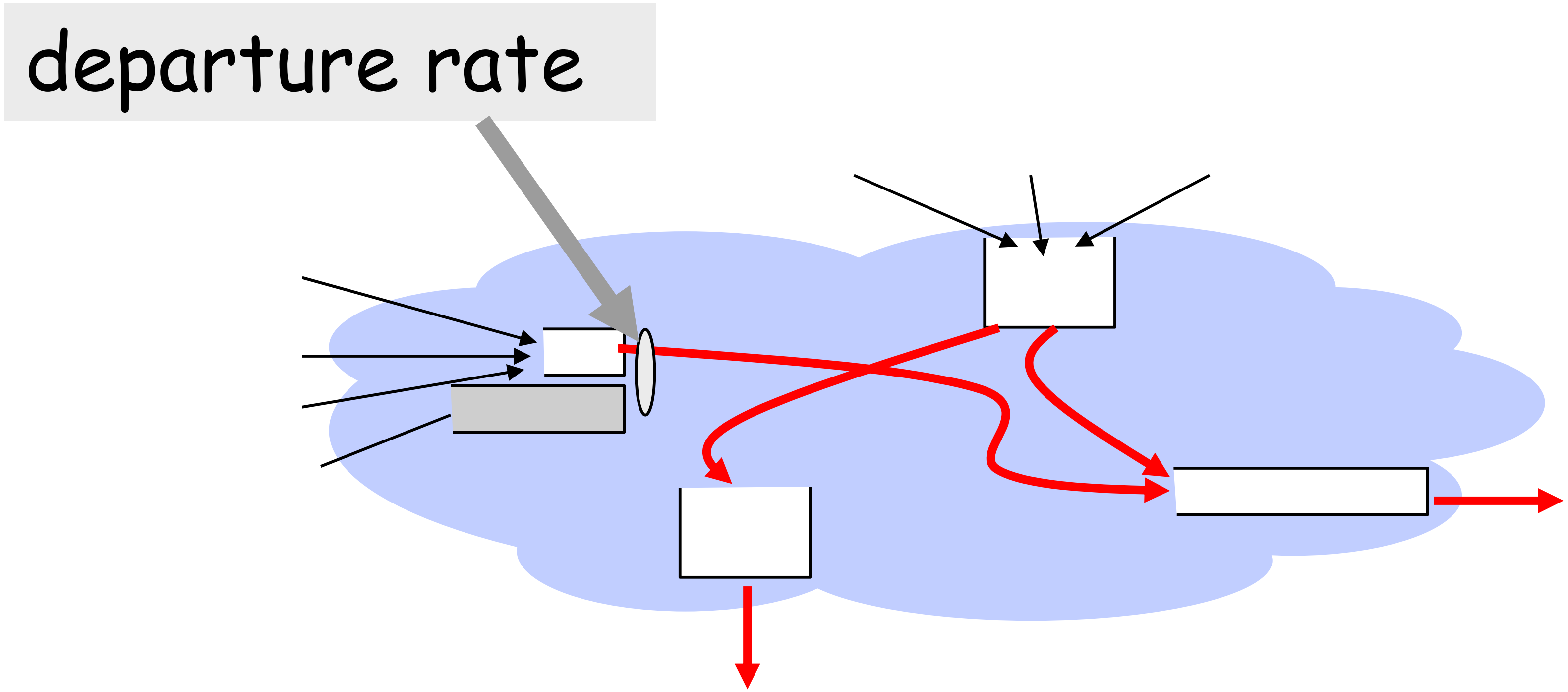
Expedited Forwarding is a building block for Diff-Serv



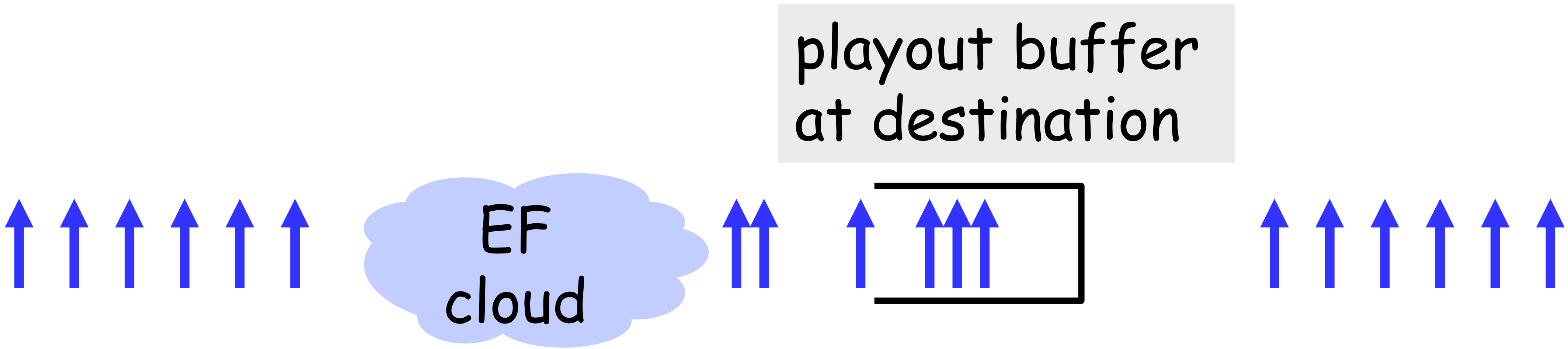
- DiffServ uses aggregate treatment of flows (for scalability)
- shaping at edge + aggregate scheduling
- ≈ priority queue
- used to build « Virtual Wire », a service similar to ATM CBR

Specification of EF : RFC 2598, June 1999

- for one EF aggregate: departure rate $\geq r$
  - measured over any interval $\geq$ 1 packet
  - r is the *configured rate* at one EF node



The old EF specification is used in Virtual Wire drafts to bound delay jitter



- Virtual Wire drafts conclude that
  - end-to-end delay jitter $\leq \alpha$ T
    α = utilization factor, T = interval at source

The Virtual Wire jitter bound contradicts other known results

- jitter bounds are hard to find for edge shaping + aggregate scheduling
- existence of finite bounds in the general case is still an open problem
  - Lu and Kumar 1991, Rybko and Stolar 1992, Seidmann 1994, Bramson 1994, Andrews 2000
- Andrews 2000 presents an example of a network which is *unstable* for some $\alpha<1$

Closed form bounds for delay

- if $\alpha \leq 1/(h-1)$ there is a closed form bound

$$D = h \frac{e+\tau}{1-(h-1)\alpha}$$

h = number of hops, α = utilization factor

- the bound diverges for $\alpha \to 1/(h-1)$
- compare to virtual wire bound α T
- if $\alpha > 1/(h-1)$, for any x, there is a network where the worst case jitter $\geq$ x
- this contradicts the Virtual Wire bound

Derivation of the bound

- Assume nodes are GR (orFIFO-per aggregate rate latency service curve elements)

1) Assume delay bound hD on low delay traffic (EF) exists, where h = max number of hops, D = max delay bound per node
2) An arrival curve of aggregate traffic at node i

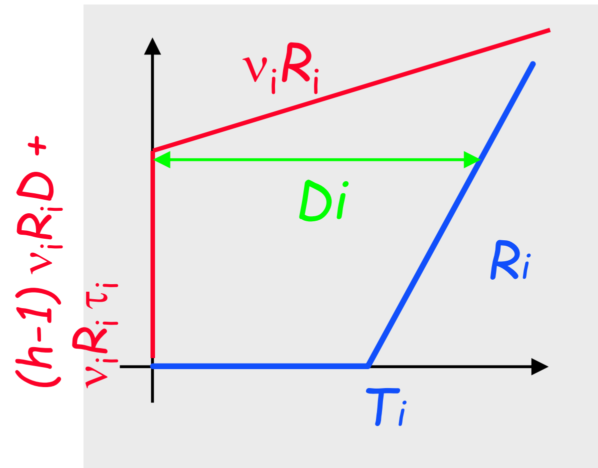
$$\alpha_i(t) = \Sigma_{m \ni i} (r_m t + (h-1) r_m D + b_m) = \nu_i R_i t + (h-1)\, \nu_i R_i D + \nu_i R_i\, \tau_i$$

where $\nu_i = (\Sigma_{m \ni i}\, r_m)/R_i$ and $\tau_i = (\Sigma_{m \ni i}\, b_m)/\,(\Sigma_{m \ni i}\, r_m)$

3) Compute horizontal distance between $\alpha_i(t)$ and $\beta_i(t)$:

$$D_i = T_i + (h-1)\, \nu_i D + \nu_i \tau_i$$

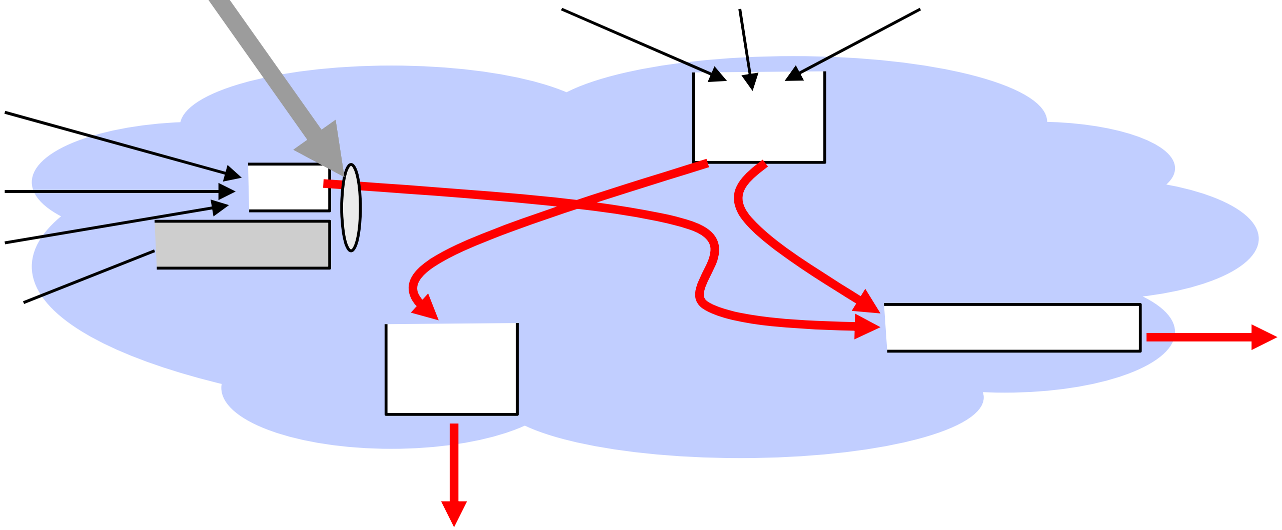
4) Deduce $D \le (T + \nu\tau)/(1 - (h-1)\nu)$ where $T = \max_i T_i$, $\nu = \max_i \nu_i$ and $\tau = \max_i \tau_i$
5) Show that finite bound exists at any time t, and let t -> ∞



The contradiction is in the specification of EF

- for practically all known nodes, the EF condition is not true
  - jitter and source rate fluctuations

departure rate may be > r



Another specification is needed

- should allow delay and backlog computations
- should apply well to reasonable routers
  - combinations of schedulers, queue, delay elements
  - basic schedulers should be easy to model
  - concatenation
- at the 49th IETF (Dec 2000, San Diego), the old EF specification is abandoned in favor of a new one based on packet scale rate guarantee

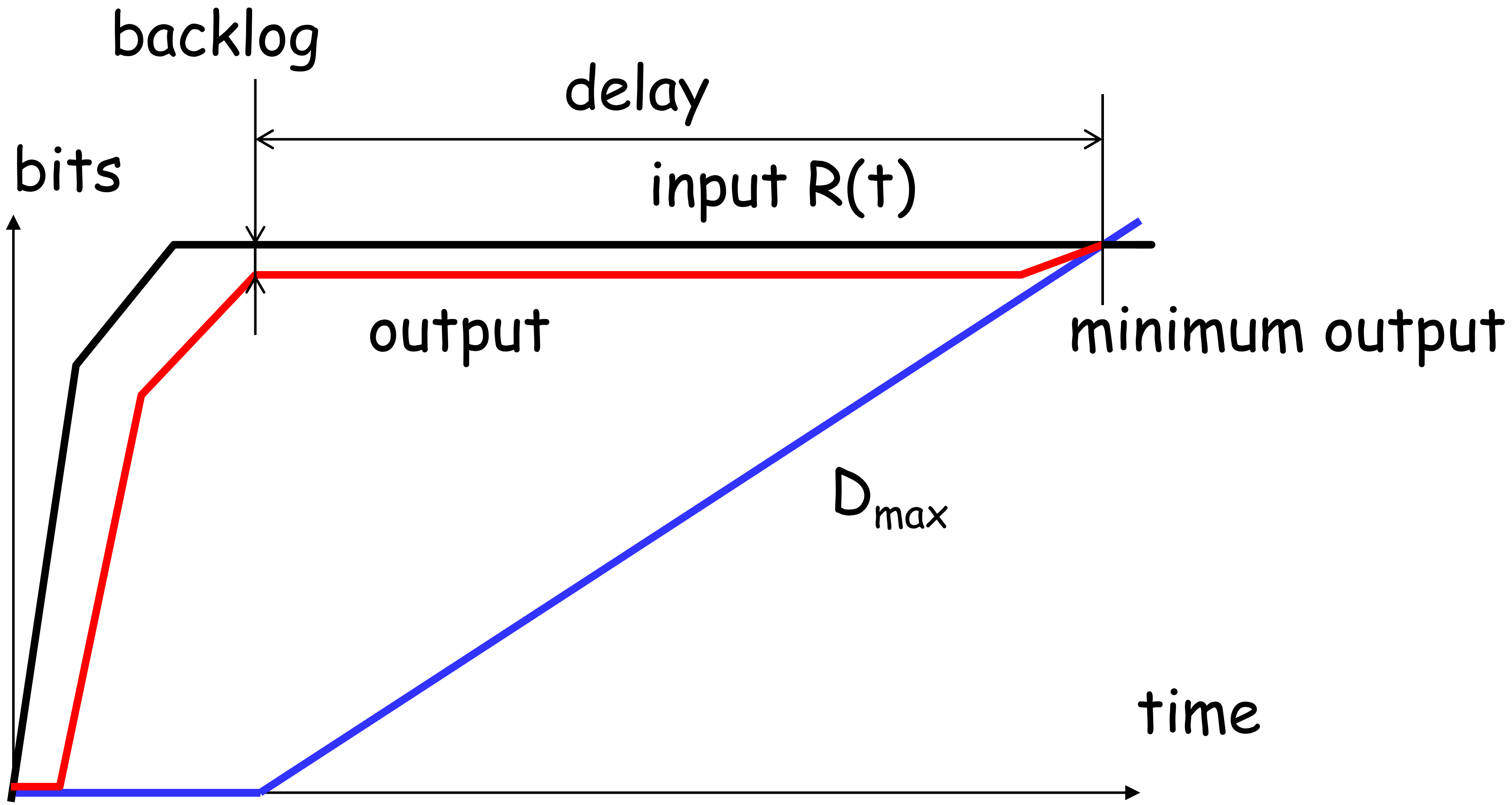
Why not use GR as a node model ?

- has all nice properties seen before: bounds, concatenation
- but: delay-from-backlog bound

given observed backlog is B, delay ?

- why ?
  - we want to control delay from backlog
  - diff-serv is not loss-free
  - if a network element has a small buffer, it should guarantee a low delay

GR node does not support a backlog-from-delay bound

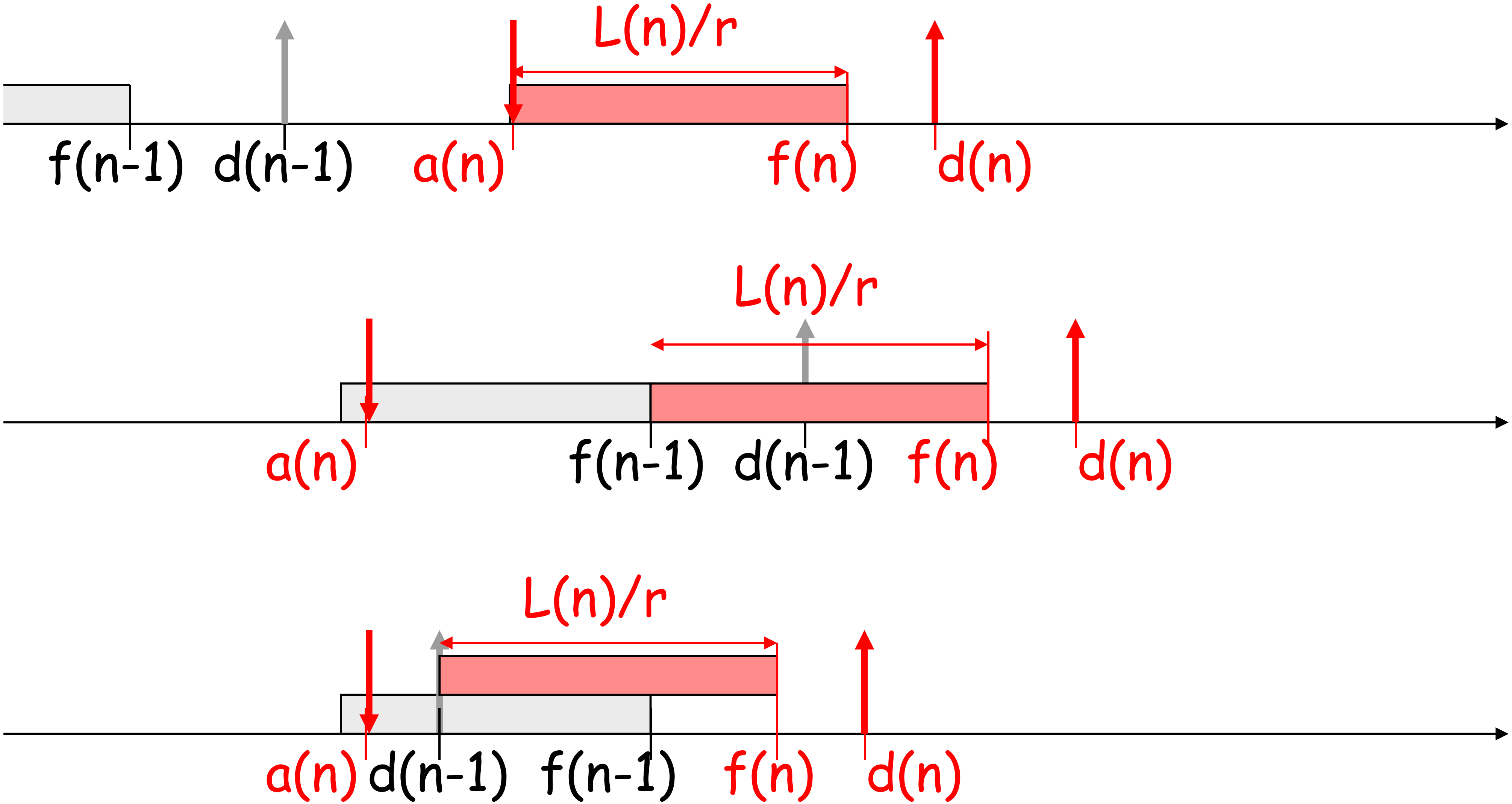


Packet Scale Rate Guarantee is the definition used for EF

- $d(n) \leq f(n) + e$
  $f(n) = \max\{a(n), \min[d(n-1), f(n-1)]\} + L(n)/r$

  $d(n)$: departure time for packet n
  $a(n)$: arrival time
  $f(n)$: virtual finish time, $f(0) = 0$

$$f(n) = \max\{a(n), \min[d(n-1), f(n-1)]\} + L(n)/r$$



PSRG has all the nice properties

- priority scheduler: $r=C$, $e=L_{max}/C$
- packet based GPS, with accuracy E_1, E_2:

$$G(n) - E_1 \leq d(n) \leq G(n) + E_2$$

G(n) = departure time in fluid GPS system

$\Rightarrow$ rate r, $e = E_1 + E_2$

- concatenation of FIFO nodes: same as GR

Delay from Backlog

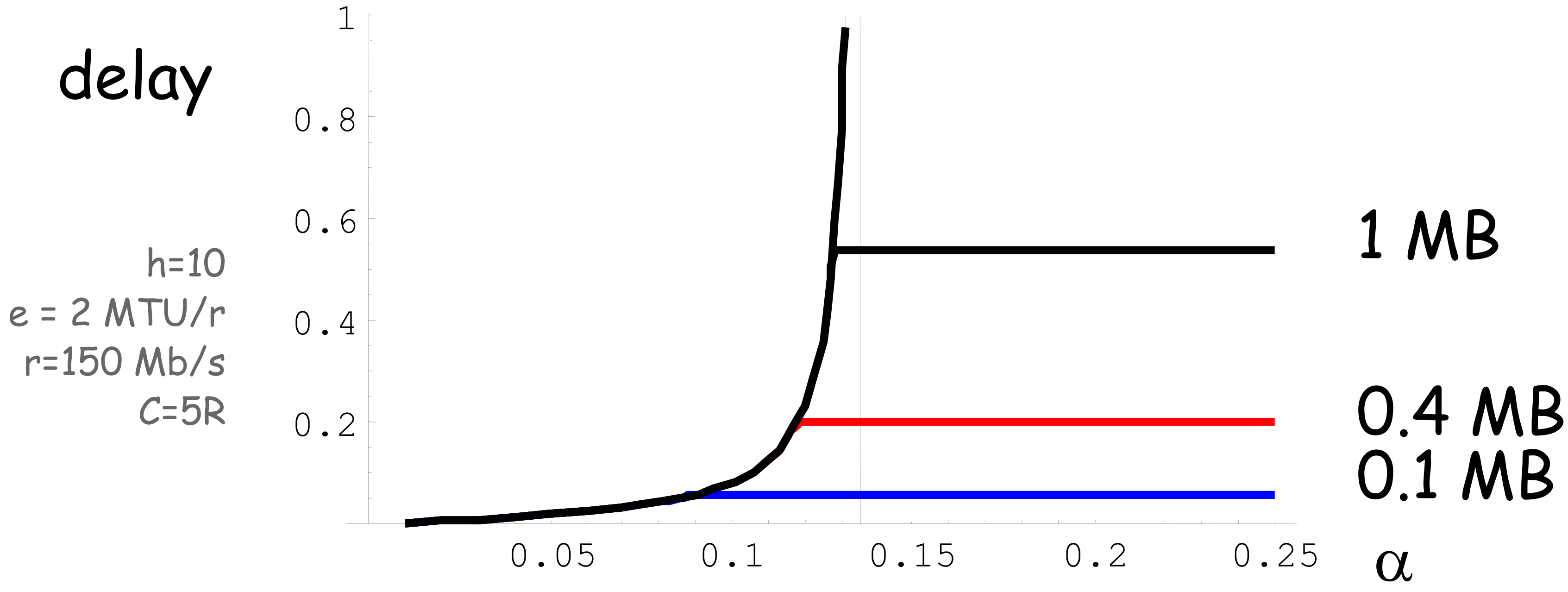
- *Theorem :*
  packet scale rate guarantee
  $\Rightarrow$ delay $\leq$ Q/r + e

  Q: backlog upon arrival

- intuitively clear -- and proof is simple -- if node is FIFO (infocom 2001)

Delay bounds

- We can combine all results above and find finite and infinite buffer bounds



PSRG versus Service Curve

- PSRG => GR
  - (but not conversely !)

- thus PSRG(r,e) => service curve (r, e+l_{max}/r)

- there are identical relations
  - PSRG <-> *adaptive* service curve (Cruz, 1998)
  - GR <-> service curve

A Min-Max approach to solve the non-FIFO case

- routers are FIFO per flow
  - all OK with IntServ (per-flow scheduling)
- EF use aggregate scheduling
  - routers are *not* FIFO per aggregate
- establishing the properties of PSRG with non-FIFO nodes has been an open challenge
- a Min-Max approach can break it

- we can get rid of f(n) by solving

(1) $$f(j)=\max\left[a(j),\min\left(d(j-1),f(j-1)\right)\right]+\frac{L(j)}{r}$$

- define
$$\begin{cases} F_j := f(j)+\dfrac{L(j+1)+\ldots+L(n)}{r} \\ A_j := a(j)+\dfrac{L(j)+\ldots+L(n)}{r} \\ D_j := d(j)+\dfrac{L(j+1)+\ldots+L(n)}{r} \end{cases}$$

- we obtain

(2)
$$F_j=\max\left[A_j,\min\left(F_{j-1},D_{j-1}\right)\right]$$

The beauty of algebra

(2) $$F_j = \max\left[A_j, \min\left(F_{j-1}, D_{j-1}\right)\right]$$

- re-write (2) by the replacement rule (min, max) -> (+,x) and obtain

$$F_j = A_j\left(F_{j-1} + D_{j-1}\right)$$

- use Gauss elimination

$$F_n = \sum_{j=0}^{n-1} A_n \ldots A_{j+1} D_j$$

- use the reverse replacement rule

$$F_n = \min_{j=0}^{n-1}\left\{\max\left[A_n, \ldots, A_{j+1}, D_j\right]\right\}$$

Alt. Characterization of PSRG

- a node is PSRG(r,e) iff
  for all n and j in [0, n-1]

(A)
$$d(n) \le e + d(j) + \frac{L(j+1) + \ldots + L(n)}{r}$$

or there is some k in [j+1,..., n] such that

(B)
$$d(n) \le e + a(k) + \frac{L(k) + \ldots + L(n)}{r}$$

- interpretation: replaces VJ's intuition

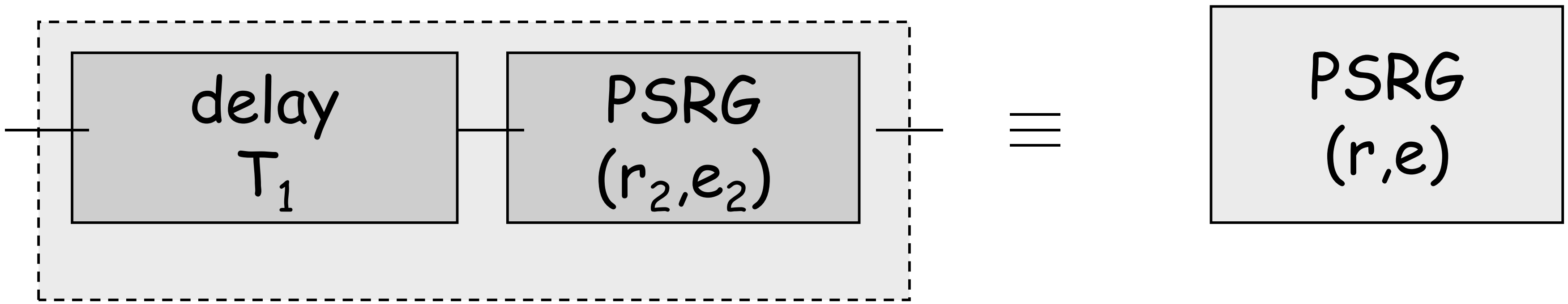
Applications

- *Theorem :*
  packet scale rate guarantee
  $\Rightarrow$ delay $\leq$ Q/r + e

  holds also for non-FIFO nodes

PSRG has all the nice properties...

- ... but concatenation results for non-FIFO nodes are harder to get
- (Le Boudec and Charny 2001):



$$r = r_2$$

$$e = e_2 + T_1 + \min\left\{ \sup_{0\le t}\left[\frac{\alpha(t+T_1) - L_{\min}}{r} - t\right], \sup_{0\le t\le T_1}\left[\frac{\alpha(t)+\alpha(T_1) - 2L_{\min}}{r} - t\right]\right\}$$

SETF

- an alternative to EF [Zhi-Li Zhang 2000]
- leads to a worst case bound which is finite for all $\alpha<1$
- packets stamped with arrival time at network access
  - aggregate scheduling
  - inside aggregate, order is that of timestamps

Theorem:

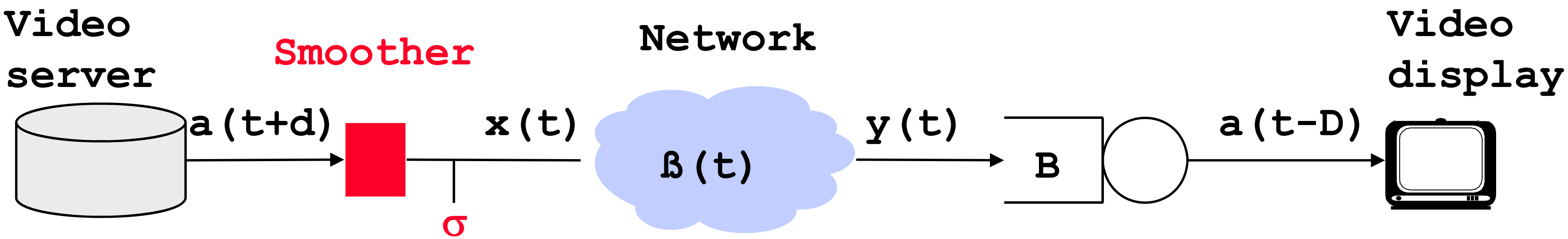
$$D = (e+\tau)\frac{1-(1-\alpha)^h}{\alpha(1-\alpha)^{h-1}}$$

Proof: similar to previous bound

Contents

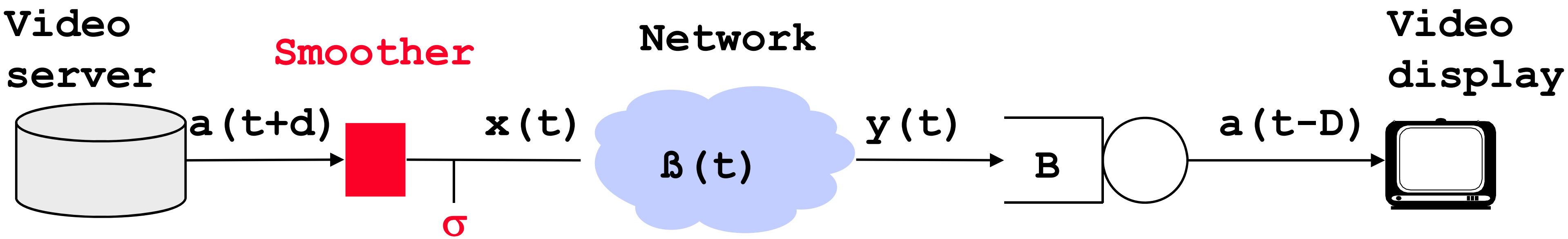
1. Arrival curves
2. Service curves, backlog, delay bounds
3. Diffserv: intuition and formal definition behind EF
4. Playback Delay for pre-recorded video
5. Statistical multiplexing with EF

Network delivery of Pre-recorded video



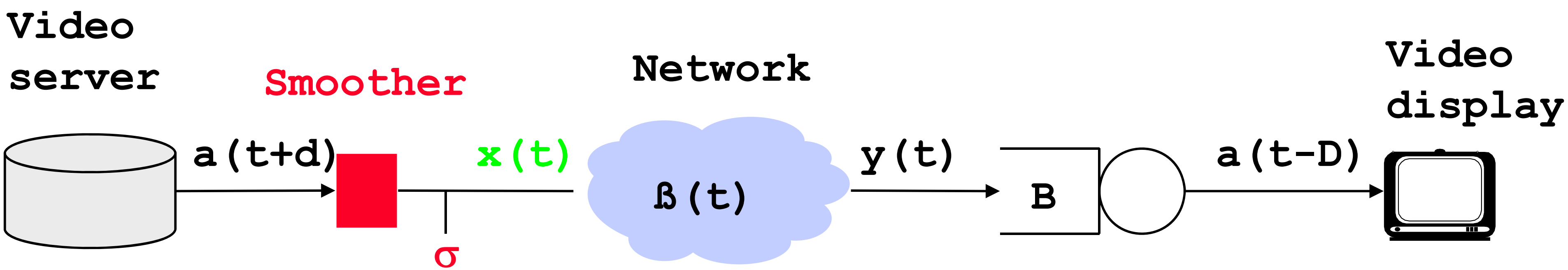
- Le Boudec and Verscheure ToN 2000, Thiran, Le Boudec and Worm, Infocom 2001
- Network + end-station offers a service curve β to flow *x(t) (intserv or diffserv + real time model of end-station)*
- Smoother delivers a flow *x(t)* conforming to an arrival curve σ. Can look-ahead on the server (max *d* time units)
- Video stream is stored in the client buffer *B* and read after a playback delay *D*.

Network delivery of Pre-recorded video



- What are the minimal values of D and B, given d, σ and β ?
- What is the scheduling (smoothing) strategy at the sender side that achieves these minimal values ?
- Is this optimal smoothing strategy unique ?
- Does a large look-ahead delay d help in reducing D and B ?

Putting the Problem into Equations



- Smoothed flow $x(t)$ such that

  $x(t) \le \delta_0(t)$ (i.e, $x(t) = 0$ if $t \le 0$)

  $x(t) \le a(t+d)$ (look-ahead up to d time units)

  $x(t) \le (x \otimes \sigma)(t)$ (smoothing)

- Output flow $y(t)$ such that

  $y(t) \ge a(t-D)$ (no buffer underflow)

  $y(t) \le a(t-D) + B$ (no buffer overflow)

- $y(t) = \Pi(x)(t)$ is not known but $(x \otimes \beta)(t) \le y(t) \le x(t)$

The Min-Plus Residuation Theorem

- From Baccelli et al, "Synchronization and Linearity"
- Theorem: Assume that the operator Π is isotone and upper-semi-continuous.
  the problem
  $x(t) \le a(t) \wedge \Pi(x)(t)$
  has one maximum solution, given by $x(t) = \underline{\Pi}(a)(t)$
- (Definition of closure of an operator)
  $$\underline{\Pi}(x) = \inf \{x, \Pi(x), \Pi \circ \Pi(x), \Pi \circ \Pi \circ \Pi(x), \ldots\}$$
- Π is isotone if $x(t) \le y(t)$ -> $\Pi(x)(t) \le \Pi(y)(t)$
- Π is upper-semi continuous if $\inf_i(\Pi(x_i)) = \Pi(\inf_i(x_i))$
  - true in practice for all our systems
- The greedy shaper output is an example of use

Massaging the Equations to use Residuation

- Output flow $y(t)$ such that

  $(x \otimes \beta)(t) \geq a(t-D)$ (no buffer underflow)

  $x(t) \leq a(t-D) + B$ (no buffer overflow)

  or equivalently using deconvolution operator $\oslash$

  $x(t) \geq (a \oslash \beta)(t-D) = \sup_u \{ a(t-D+u) - \beta(u) \}$

  $x(t) \leq a(t-D) + B$

- Therefore find smallest D, B s.t. maximal solution of

  $x(t) \leq \{ \delta_0(t) \wedge a(t+d) \wedge (a(t-D) + B) \} \wedge \{(x \otimes \sigma)(t) \}$

  verifies

  $x(t) \geq (a \oslash \beta)(t-D)$

Applying Residuation to our Problem

- Maximal solution of

$$x(t) \le \{ \delta_0(t) \wedge a(t+d) \wedge (a(t-D) + B) \} \wedge \{(x \otimes \sigma)(t) \}$$

is, with σ sub-additive,

$$x(t) = \sigma \otimes \{ \delta_0(t) \wedge a(t+d) \wedge (a(t-D) + B) \}$$
$$= \sigma(t) \wedge \{ (\sigma \otimes a)(t-D) + B \} \wedge (\sigma \otimes a)(t+d)$$

- Need to check that this solution $x(t) \ge (a \oslash \beta)(t-D)$
  - $\sigma(t) \ge (a \oslash \beta)(t-D)$

    -> $D \ge h(a, \beta \otimes \sigma)$
  - $(\sigma \otimes a)(t-D) + B \ge (a \oslash \beta)(t-D)$

    -> $B \ge v(a \oslash a, \beta \otimes \sigma)$
  - $(\sigma \otimes a)(t+d) \ge (a \oslash \beta)(t-D)$

    -> $D + d \ge v(a \oslash a, \beta \otimes \sigma)$

Bounds for D, B and d

- In summary, we have shown that
  - the set of admissible playback delays D, playback buffer B and look-ahead limit d is

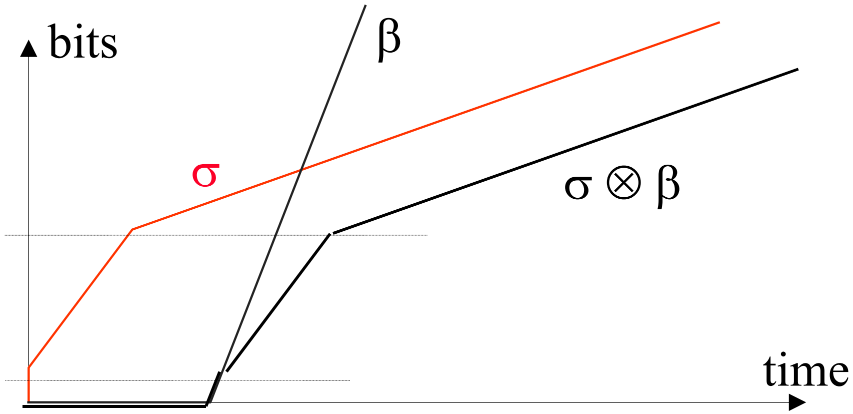
    $D \geq D_{min} = h(a, \beta \otimes \sigma)$

    $D + d \geq (D+d)_{min} = h(a \oslash a, \beta \otimes \sigma)$

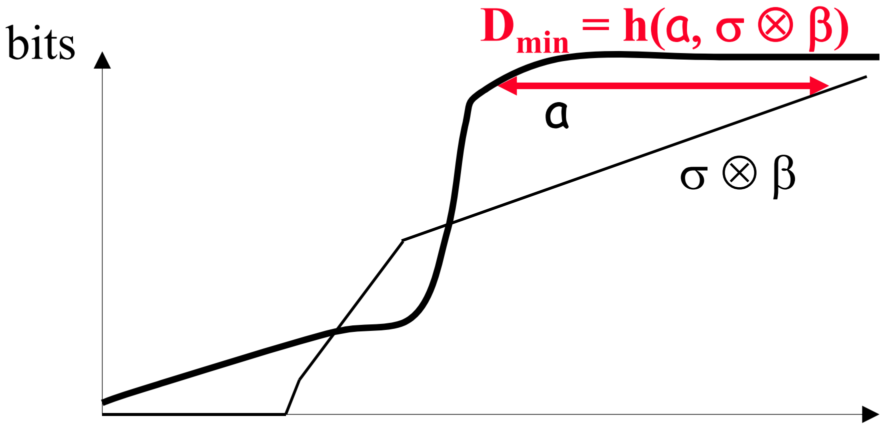
    $B \geq B_{min} = v(a \oslash a, \beta \otimes \sigma)$
  - in particular, there is a minimum playback delay.
  - if D, d, B satisfy the constraints above, a schedule is possible;
    else, there is no schedule that can guarantee correct operation

The formulae have a simple graphical interpretation

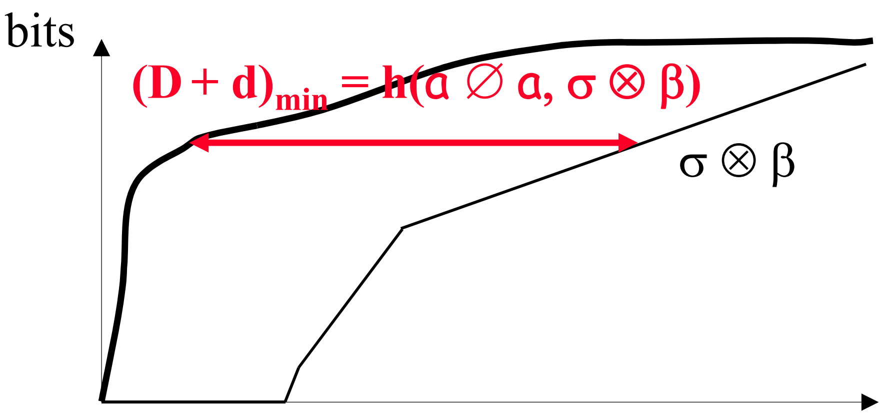
(1) compute $\sigma \otimes \beta$



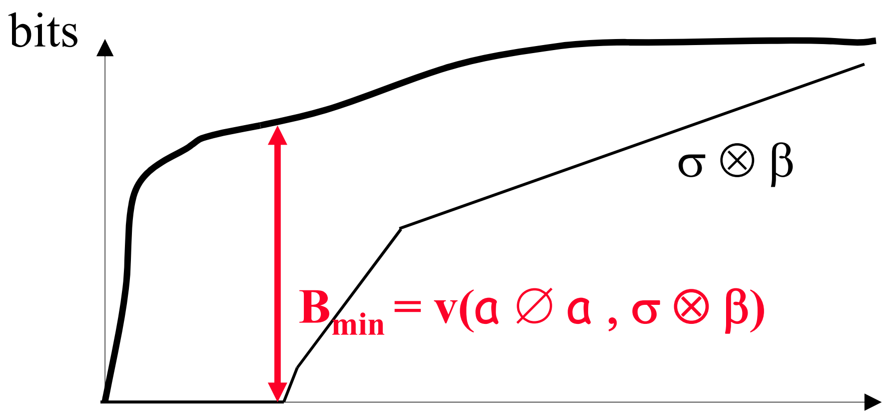
(2) compute the horizontal deviation



(3) compute $\alpha \oslash \alpha$ and the horizontal deviation

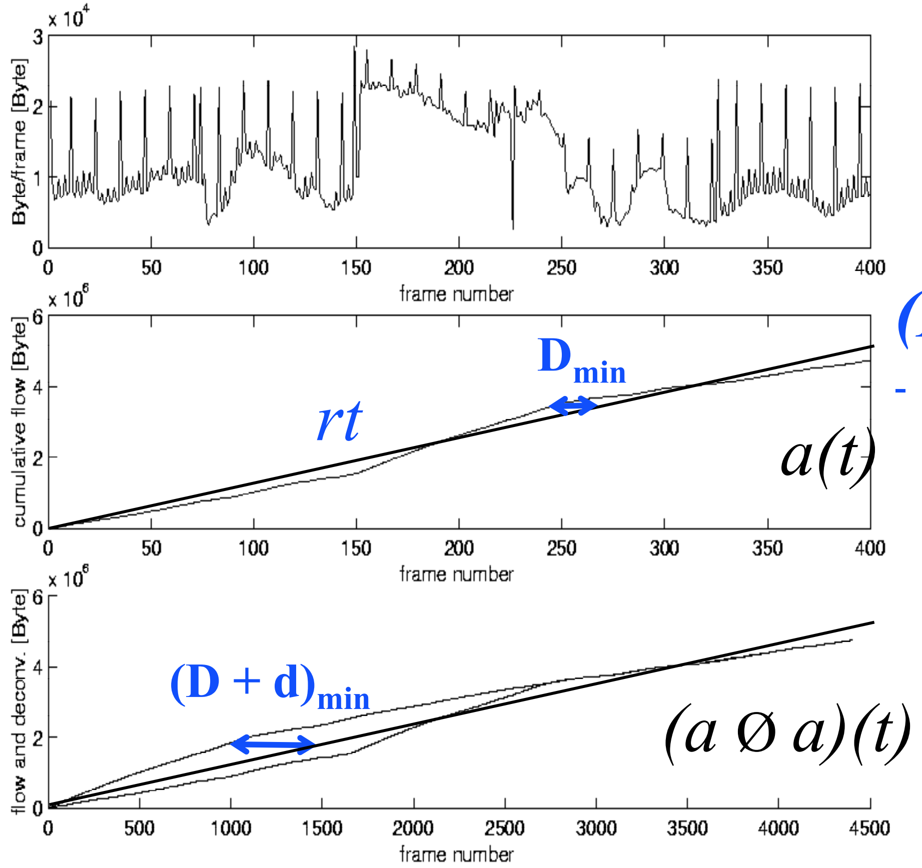


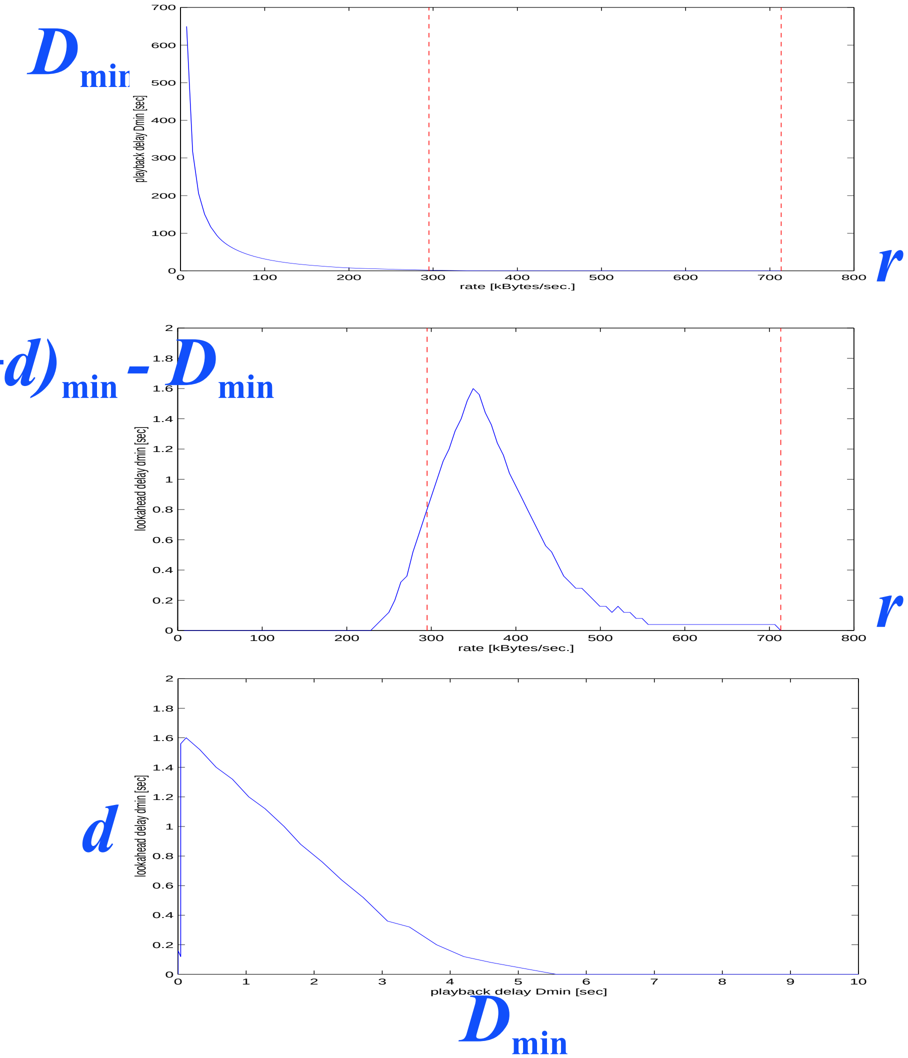
(4) compute the vertical deviation



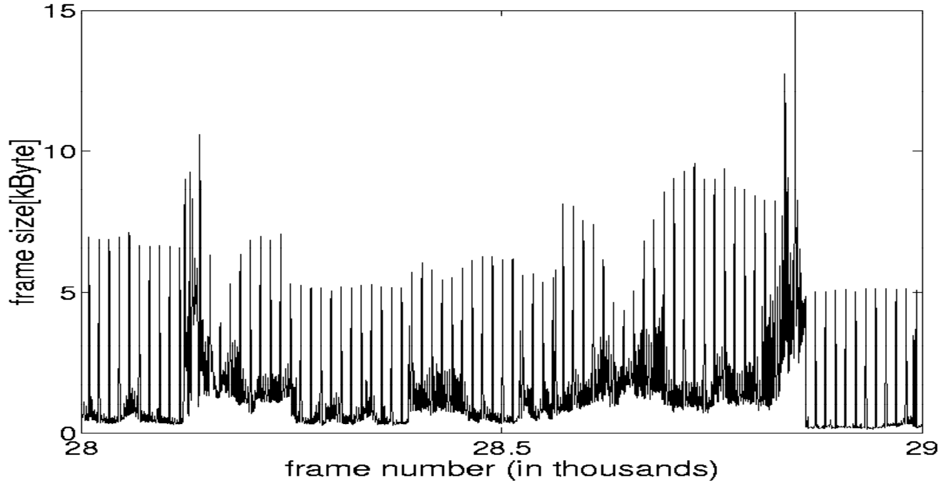
Example: MPEG Trace

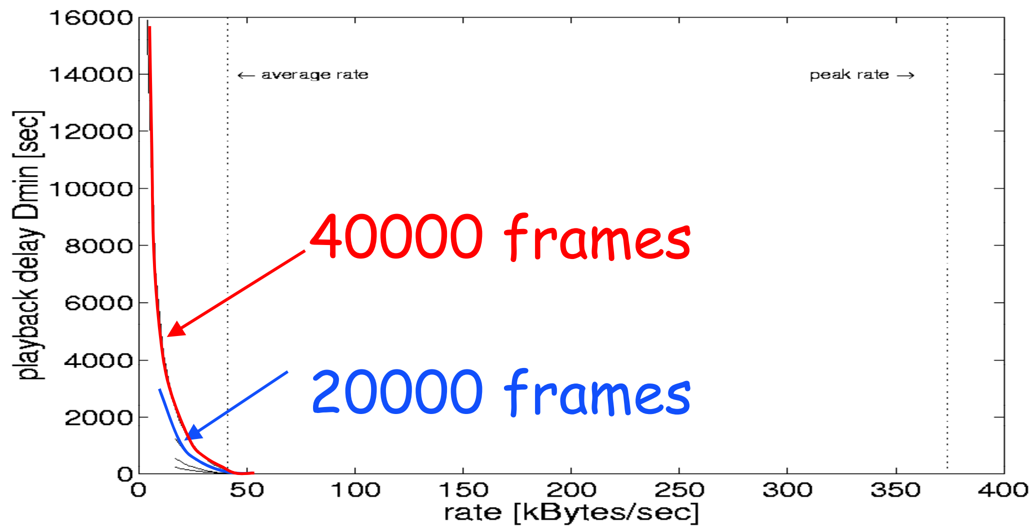
- MPEG files, 25 frames/sec, discretized in packets of 416 bytes

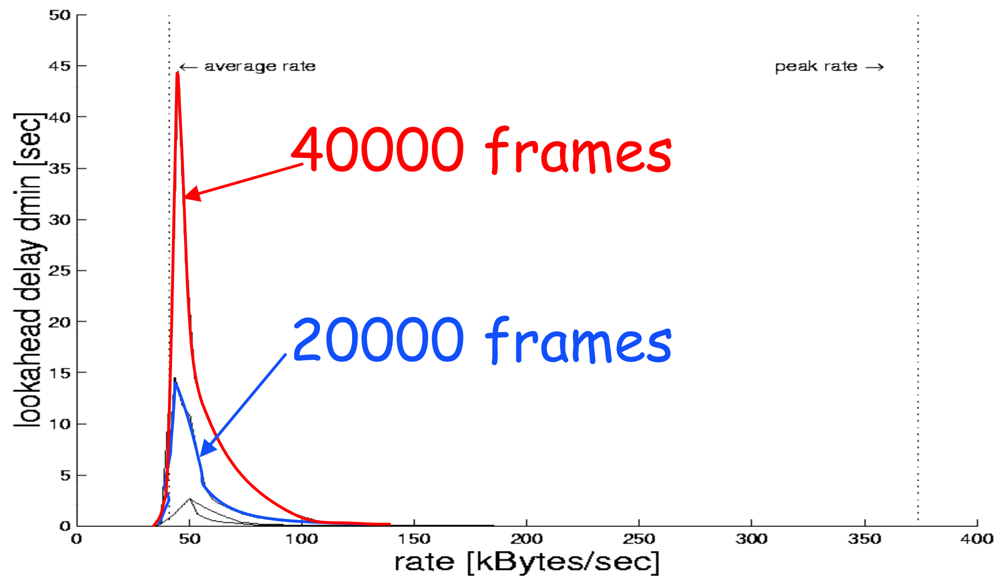




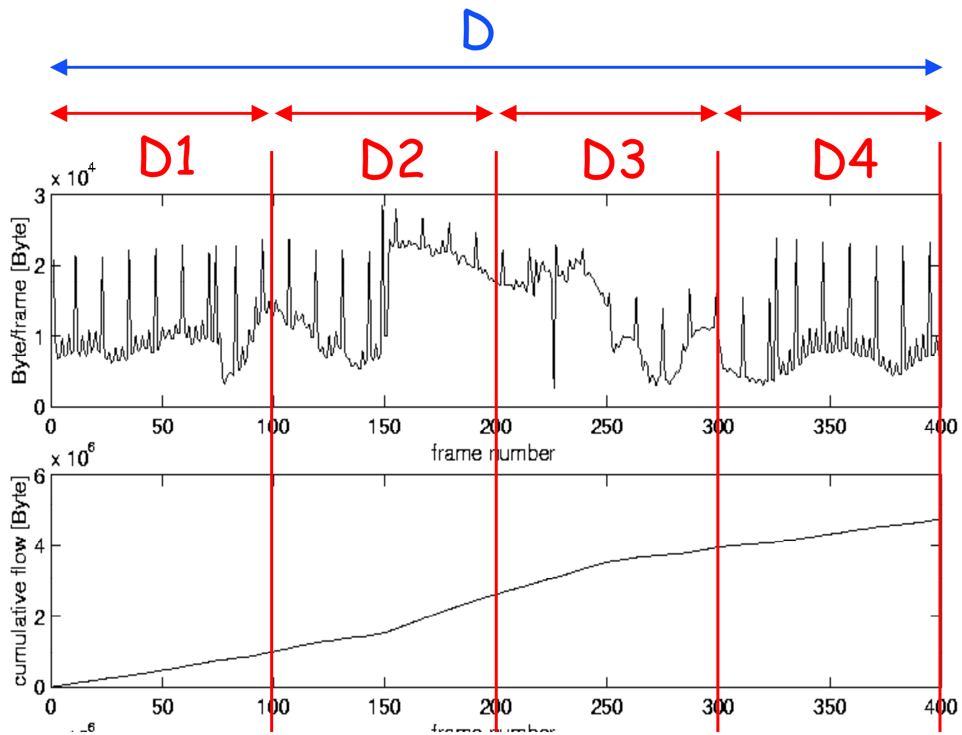
- Actual values of delays depend on the length of the stream and the position of largest burst, and the ability to predict it
  - Example: in Jurassic Park trace, largest burst occurs between frames 28000 - 29000

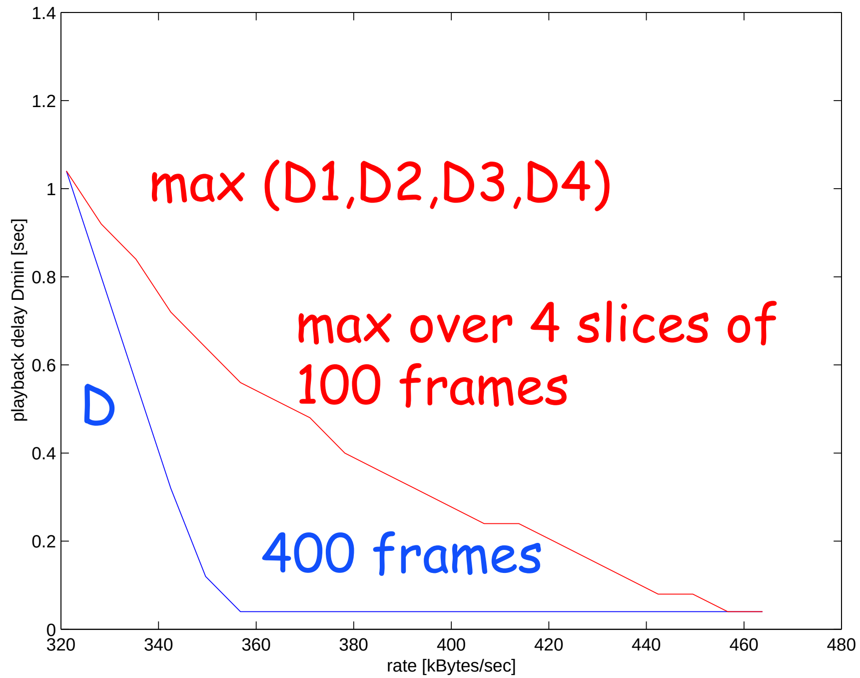






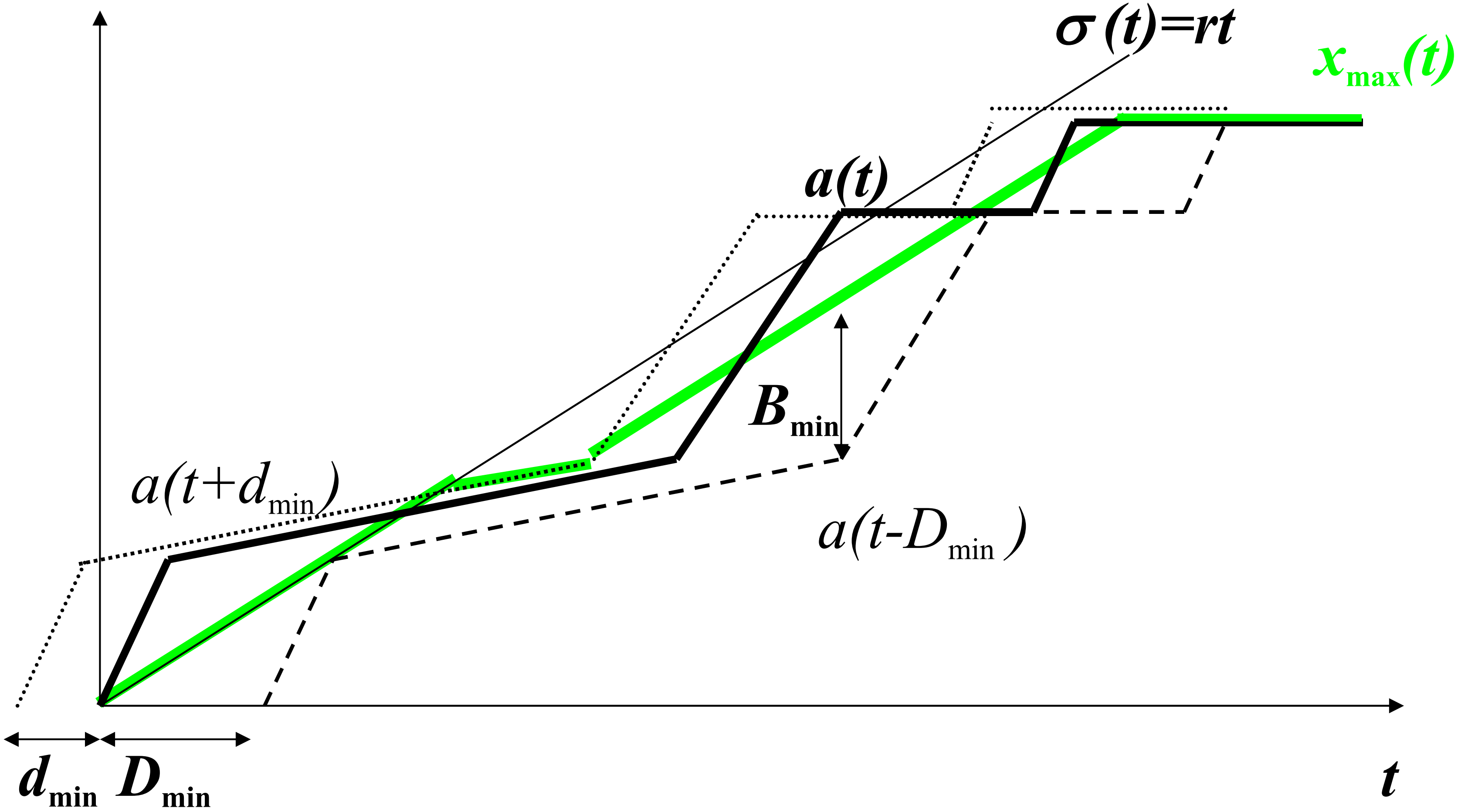
- Actual values of delays depend on the length of the stream, the position of largest burst, and the ability to predict it





Scheduling for D_{min}, d_{min} and B_{min}

$$x_{max}(t) = \sigma(t) \wedge (\sigma \otimes a)(t+d_{min}) \wedge \{ (\sigma \otimes a)(t-D_{min}) + B_{min} \}$$



Example 3: Dual problem formulation

- Find smallest D, B and d s.t. the maximal solution of

  $$x(t) \le \delta_0(t) \wedge R(t+d) \wedge \{R(t-D) + B\} \wedge (x \otimes \sigma)(t)$$

  verifies

  $$x(t) \ge (R \oslash \beta)(t-D) .$$

- Property of $\oslash$: $x \le (x \otimes \sigma)$ <-> $(x \oslash \sigma) \le x$

- Find smallest D, B and d s. t. the minimal solution of

  $$x(t) \ge (R \oslash \beta)(t-D) \vee (x \oslash \sigma)(t)$$

  verifies

  $$x(t) \le \delta_0(t) \wedge R(t+d) \wedge \{R(t-D) + B\} .$$

Max-Plus System Theory in Action

- From Baccelli et al, "Synchronization and Linearity"; assume that Π is isotone and lower-semi-continuous.

  Theorem : the problem

  $$x(t) \le a(t) \vee \Pi(x)(t)$$

  has one minimum solution, given by $x_{min}(t) = \overline{\Pi}(a)(t)$

- (Definition of super-additive closure)

  $$\overline{\Pi}(x) = \sup \{x, \Pi(x), \Pi \circ \Pi(x), \Pi \circ \Pi \circ \Pi(x), \ldots\}$$

- Minimal solution of

  $$x(t) \geq (R \oslash \beta)(t-D) \vee (x \oslash \sigma)(t)$$

  is, with σ sub-additive with $\sigma(0) = 0$,

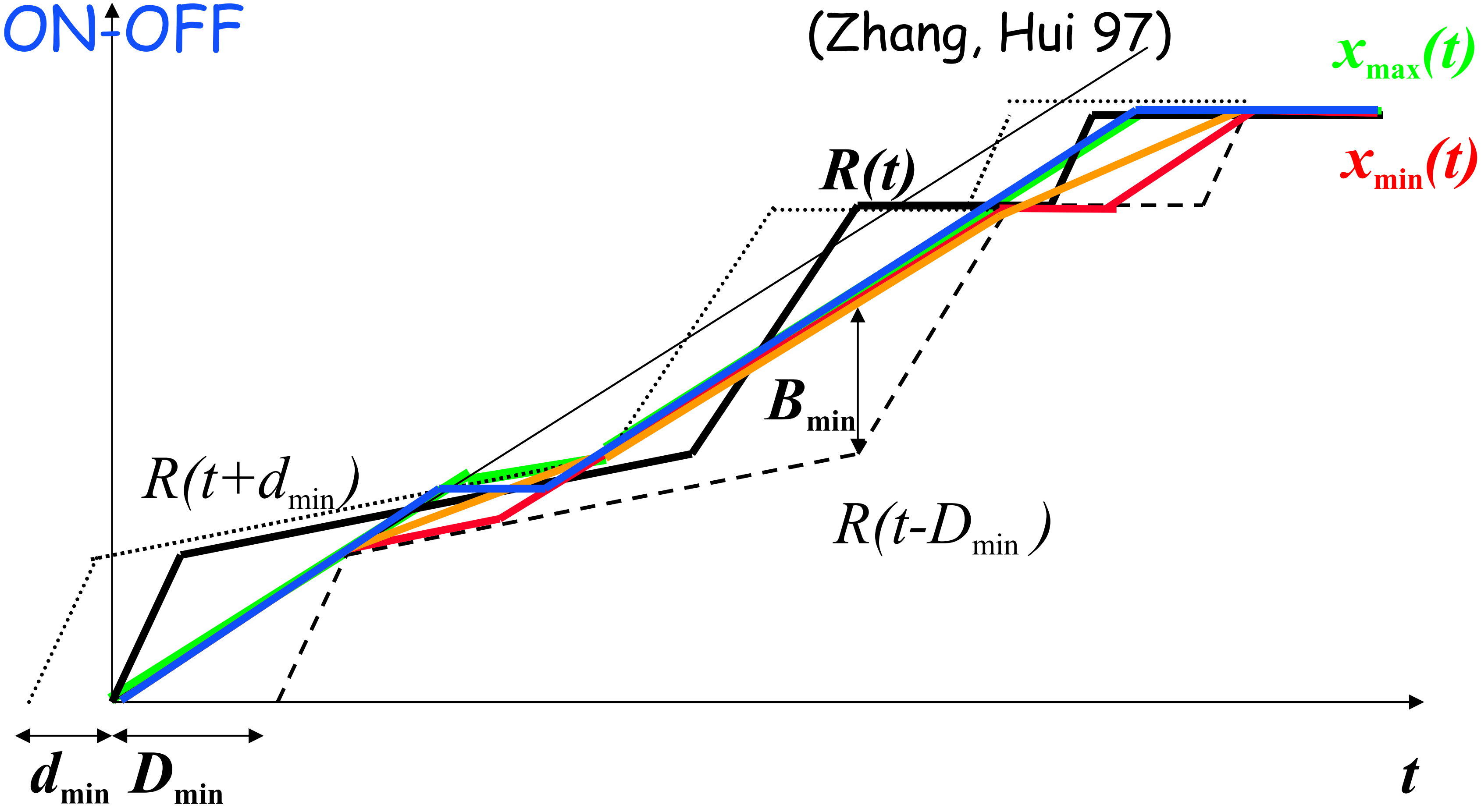
  $$x_{min}(t) = (R \oslash (\beta \otimes \sigma))(t-D)$$

Scheduling for D_{min}, d_{min} and B_{min}

$x_{max}(t) = \sigma(t) \wedge (\sigma \otimes R)(t+d_{min}) \wedge \{ (\sigma \otimes R)(t-D_{min}) + B_{min} \}$

$x_{min}(t) = (R \oslash (\beta \otimes \sigma))(t-D)$ (Le Boudec, Verscheure 2000)

+ Other metrics (Feng, Rexford 99):
+ minimal rate variability (Salehi, Zhang, Kurose, Towsley 98)
+ ON-OFF (Zhang, Hui 97)



Contents

1. Arrival curves
2. Service curves, backlog, delay bounds
3. Diffserv: intuition and formal definition behind EF
4. Min-plus algebra in action: Video smoothing
5. Statistical multiplexing with EF

5. Stochastic Bounds

- network calculus gives deterministic bounds on delay and loss
- combine with Hoeffding bounds [1963]: **Assume**
  - X_i are independent and $0 \leq X_i \leq 1$
  - $E(X_1 + \ldots + X_I) = s$ is known

  then for $s < x < I$

$$P(X_1 + \ldots + X_I > x) \leq \exp -\left(x \ln \frac{x}{s} + (I - x) \ln \frac{I - x}{I - s} \right)$$

Bound on loss probability

- I independent, stationary sources with identical constraints σ_i served in a network element with ***super-additive*** service curve β [Chang, Vojnovic and L]

$$\boxed{P(Q > b) \le \inf_{\underline{s}} \left\{ \sum_k g(s_k, s_{k+1}) \right\}}$$

where $0 = s_0 < s_1 < \ldots < s_K = \tau$, $\tau = \inf\{t: \alpha(t) \le \beta(t)\}$

and for $\alpha(v) - \beta(u) > b$

$$g(u,v) = \exp\left(-I\left(\frac{\beta(u)+b}{\alpha(v)} \ln \frac{\beta(u)+b}{\rho v} + \frac{\alpha(v)-\beta(u)-b}{\alpha(v)} \ln \frac{\alpha(v)-\beta(u)-b}{\alpha(v)-\rho v} \right)\right)$$

else $g(u,v) = 0$

- Step 1: reduction to horizon τ

$$Q(0) = \sup_{s \le \tau} \{A(-s,0) - \beta(s)\}$$

- Step 2:

$$Q(0) = \max_k \{\sup_{s_k \le s \le s_{k+1}} A(-s,0) - \beta(s)\}$$
$$\le \max_k \{A(-s_{k+1},0) - \beta(s_k)\}$$

- Step 3 : Hoeffding to each term

$$A(-s_{k+1},0) - \beta(s_k) = \sum_i A_i(-s_{k+1},0) - \beta_i(s_k)$$

$$A_i(-s_{k+1},0) - \beta_i(s_k) \le \alpha_i(s_{k+1}) - \beta_i(s_k)$$

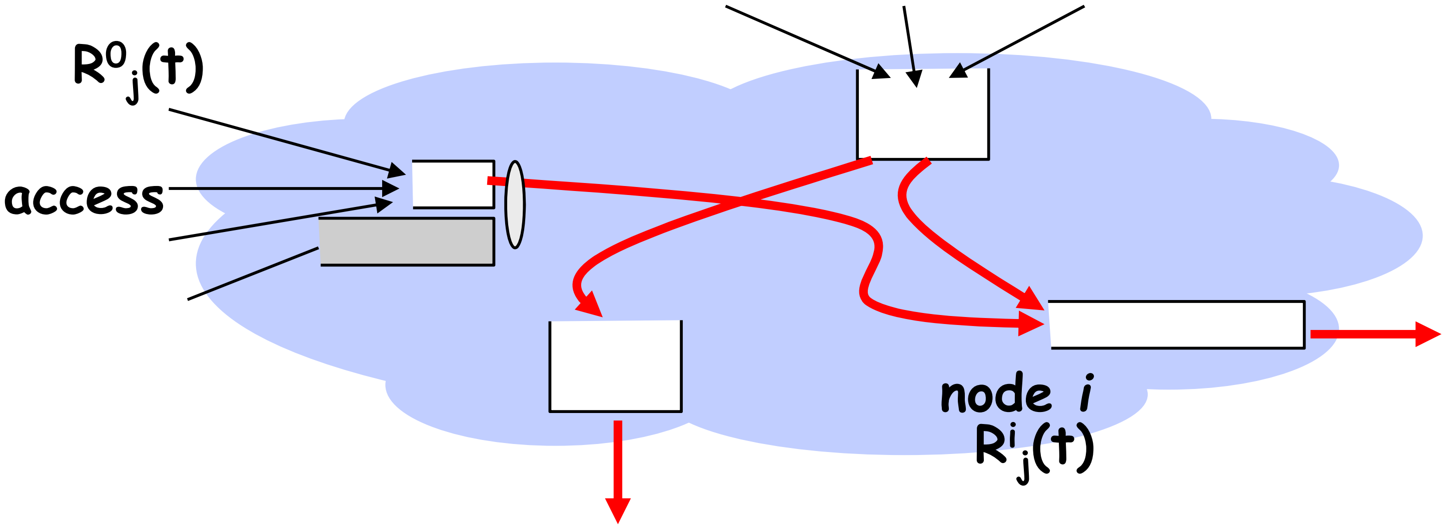
$$E\{A\ (-s_{k+1},0) - \beta(s_k)\} \le \rho s_{k+1} - \beta\ (s_k)$$

Application to DiffServ

- micro-flows in one aggregate assumed independent at network access only
- at node *i* majorize the amount of data in [s,t] by

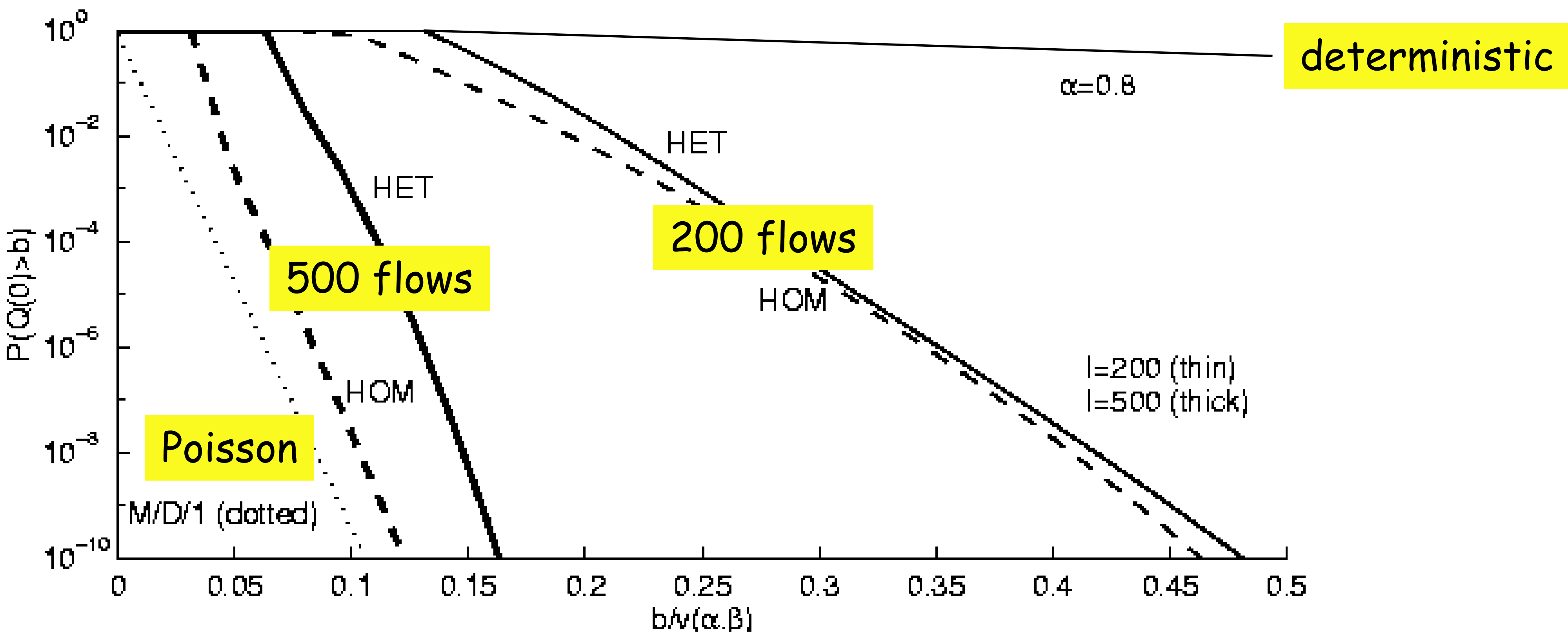
$$R^i_j(t) - R^i_j(s) \leq R^0_j(t) - R^0_j(s-d)$$

and apply the previous [Chang, Song and Siu Sigmetrics 2001, Vojnovic and Le Boudec Infocom 2002]



Compare to Poisson Approximation

- Poisson approximation proposed [Bonald, Proutière, Roberts, Infocom '01] for CBR flows
- Bound converges to Poisson for many flows and small burstiness



Conclusion

- Network Calculus is a set of tools and theories for the deterministic analysis of communication networks
- A new system theory, which applies min-plus algebra to communication networks
- Does not supersede stochastic queueing analysis, but gives new tools for analysis of sample paths
- "Network calculus", J-Y Le Boudec and P. Thiran, Lecture Notes in Computer Sciences vol. 2050, Springer Verlag, also available on-line at `http://lcawww.epfl.ch`