Network QoS
371-2-0213

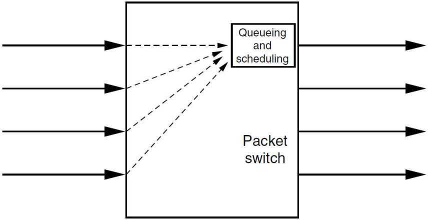
Lecture 5

Gabriel Scalosub

Outline

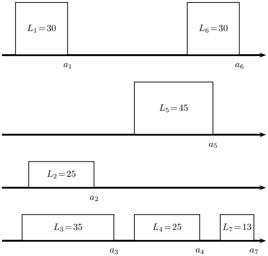
1. Network Calculus – Basics
   - Architecture and Notation
   - Reich's Equation
2. Network Calculus – Defining a New Calculus
   - Min-Sum Convolution
   - Service Curves
   - Envelopes and Regulators
3. Network Performance and Design
   - Provisioning and Delay Bounds
   - Summary

Switch Architecture



- input queues vs. *output queues*
- output queue + scheduling policy = *packet multiplexer*

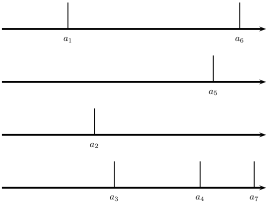
Switch Architecture



- input queues vs. *output queues*
- output queue + scheduling policy = *packet multiplexer*
- inter-arrival times
  - packet length L
  - arrival starts at time t
  - link of capacity C

$\Rightarrow$ arrival ends at time $t + \frac{L}{C}$

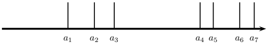
Switch Architecture



- input queues vs. *output queues*
- output queue + scheduling policy = *packet multiplexer*
- inter-arrival times
  - packet length L
  - arrival starts at time t
  - link of capacity C

  $\Rightarrow$ arrival ends at time $t + \frac{L}{C}$
- store-and-forward
  - packet *arrives* at time $t + \frac{L}{C}$

Switch Architecture

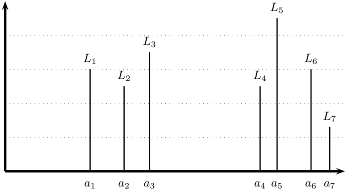


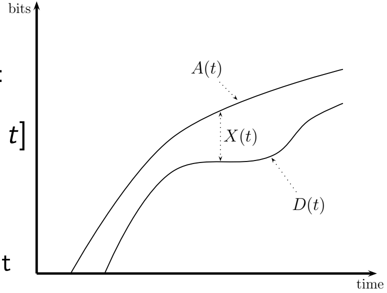
- input queues vs. *output queues*
- output queue + scheduling policy = *packet multiplexer*
- inter-arrival times
  - packet length L
  - arrival starts at time t
  - link of capacity C

  $\Rightarrow$ arrival ends at time $t + \frac{L}{C}$
- store-and-forward
  - packet *arrives* at time $t + \frac{L}{C}$

Basic Processes

- traffic:
  - a sequence of packets $p_1, p_2, \ldots$
  - arriving at an output queue
  - output link capacity C
- for each packet p_k
  - a_k: arrival time to queue
  - L_k: packet length
  - d_k: departure time from queue
- $A(t)$: cumulative arrivals in $[0, t]$
  - $A(t) = 0$ for $t < 0$
  - monotone, and right-continuous:
    - $\lim_{s \searrow t} A(s) = A(t)$
- $D(t)$: cumulative departures in $[0, t]$
  - $D(t) = 0$ for $t < 0$
  - $D(t) \leq A(t)$ for all t
- $X(t)$: queue length at t
  - includes partial packet being sent
  - $X(t) = A(t) - D(t)$ for all t



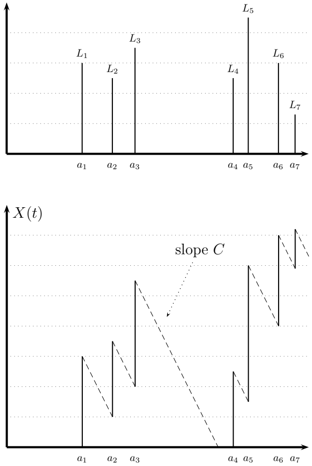


Work Conserving Schedulers

- *work conserving scheduler*
  - never idle when queue non-empty
  - buffer: empty or depletes at rate C
  - idle/busy periods
- $X(t)$ independent of scheduling policy
- busy period: same packets depart
  - *order* depends on scheduling policy
- $\overline{X}(t_1, t_2)$: time average queue size

$$\overline{X}(t_1, t_2) = \frac{1}{t_2 - t_1} \int_{t_1}^{t_2} X(u)du$$

  - also independent of scheduler
  - $\overline{X}(t_1, t_2) \sim$ average delay
- $X(t) = \sum_{\text{sessions } s} X_s(t)$
- *conservation law*
  - $\overline{X}(t_1, t_2) = \sum_s \overline{X}_s(t_1, t_2)$
  - one goes down, another must go up
  - can be controlled by the scheduler



Reich's Equation

- Notation
  - t_-: the moment just before t
  - given a function f

$$f(t_-) = \lim_{s \nearrow t} f(s)$$

- Assume:
  - work-conserving scheduler
  - outgoing link capacity C

Reich's Equation

$$X(t) = \sup_{0_- \leq s \leq t} (A(t) - A(s) - C(t-s))$$

- $A(t) - A(s)$: *actual* difference of arrivals
  - arrivals during $(s, t]$
- $C(t-s)$: *potential* service over interval of length $t - s$

Reich's Equation – Proof

Reich's Equation

$$X(t) = \sup_{0_- \leq s \leq t} (A(t) - A(s) - C(t-s))$$

Proof of Reich's Equation.

- First note that for all $s \leq t$

$$X(t) \geq A(t) - A(s) - C(t-s)$$

  - potentially, $X(s) \geq 0$
  - service during $[s, t]$ upper bounded by $C(t-s)$

$$\Rightarrow X(t) \geq \sup_{s \leq t}(A(t) - A(s) - C(t-s))$$

- $A(s) = 0$ for $s < 0$

$$\Rightarrow X(t) \geq \sup_{0_- \leq s \leq t} (A(t) - A(s) - C(t-s))$$

Reich's Equation – Proof

Reich's Equation

$$X(t) = \sup_{0_- \leq s \leq t} \left(A(t) - A(s) - C(t-s) \right)$$

Proof of Reich's Equation.

- Define: $v = \sup \{ 0_- \leq s \leq t \mid X(s) = 0 \}$
  - beginning of "t's busy period"
  - $X(v_-) = 0$
  - queue non-empty over $[v, t]$
  - depletes at rate C during $[v, t]$

  $\Rightarrow X(t) = A(t) - A(v_-) - C(t - v)$
- $A(t) - A(v_-) - C(t - v) \leq \sup_{0_- \leq s \leq t} \left(A(t) - A(s) - C(t-s) \right)$

  $\Rightarrow X(t) \leq \sup_{0_- \leq s \leq t} \left(A(t) - A(s) - C(t-s) \right)$

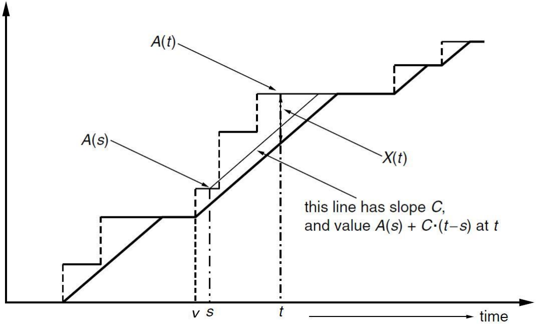
$\square$

Reich's Equation - Geometric Interpretation

- rewriting Reich's equation

$$X(t) = \sup_{0_- \leq s \leq t} (A(t) - \underbrace{(A(s) + C(t-s))}_{f_s(t)})$$

- our "goal": to push down $f_s(t)$ as much as possible



Reich's Equation - Departure Process

Corollary

$$D(t) = \inf_{0_- \leq s \leq t} (A(s) + C(t-s))$$

Proof.

$$\begin{aligned} D(t) &= A(t) - X(t) \\ &= A(t) - \sup_{0_- \leq s \leq t} (A(t) - A(s) - C(t-s)) \\ &= \inf_{0_- \leq s \leq t} (A(s) + C(t-s)) \end{aligned}$$

□

- FIFO scheduler
  - departures can be inferred from $D(t)$
  - $d_k = \inf \{u \mid D(u) \geq A(a_k)\}$

Massaging the Departure Process

- recall: $D(t) = \inf_{0_- \leq s \leq t}(A(s) + C(t - s))$
- equivalently, $D(t) = \inf_{0_- \leq s \leq t}(A(s) + C(t - s)^+)$
- since $A(s) = 0$ for $s < 0$ and $A(s)$ is non-decreasing

$$D(t) = \inf_{s \in \mathbb{R}}(A(s) + C(t - s)^+)$$

- Denote: $B(x) = Cx^+$
- We therefore have $D(t) = \inf_{s \in \mathbb{R}}(A(s) + B(t - s))$

Min-Sum Convolution

Given any two non-negative non-decreasing “real” functions $A(t)$, $B(t)$ (possibly also $+\infty$ valued), their *convolution* is defined by

$$(A * B)(t) = \inf_{\tau \in \mathbb{R}}(A(\tau) + B(t - \tau))$$

- A “real” function $A(t)$ is *causal* if $A(t) = 0$ for all $t < 0$.

Basic Observations

- for causal $A(t), B(t)$

$$(A * B)(t) = \inf_{0_- \leq \ \tau \ \leq t_+} (A(\tau) + B(t - \tau))$$

  - $\tau < 0$: $A(\tau) = 0$ (causal), $B(t - \tau) \geq B(t)$ (non-decreasing)
  - $\tau > t$: $A(\tau) \geq A(t)$ (non-decreasing), $B(t - \tau) = 0$ (causal)
  - taking $0_-, t_+$ takes care of "jumps" of $A(t), B(t)$ at $t = 0$
- Why "convolution"?
  - recall the functional analysis convolution

$$(f * g)(t) = \int_{\tau \in \mathbb{R}} f(\tau) \cdot g(t - \tau) d\tau$$

  - Sum (integral) $\longrightarrow$ Min (infimum)
  - Product $\longrightarrow$ Sum
  - distributive law maintained

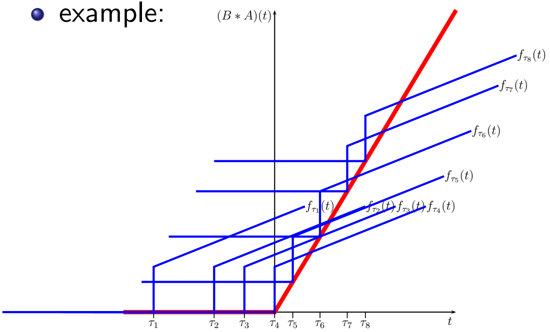
$$a \cdot (b + c) = a \cdot b + a \cdot c \quad \text{vs.} \quad a + \min\{b, c\} = \min\{a + b, a + c\}$$

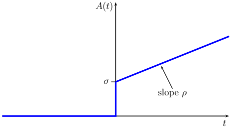
Geometric Intuition

- alternative (more geometric) view of $B * A$:
  - define $f_\tau(t) = B(\tau) + A(t - \tau)$
  - hence,

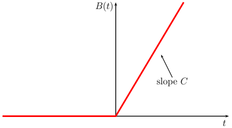
$$(B * A)(t) = \inf_{\tau \in \mathbb{R}} f_\tau(t)$$

- example:





$$A(t) = (\sigma + \rho t) I_{\{t \geq 0\}}$$



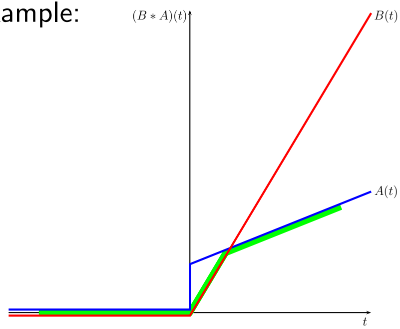
$$B(t) = Ct^+$$

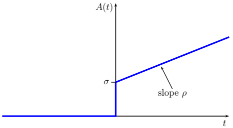
Geometric Intuition

- alternative (more geometric) view of $B * A$:
  - define $f_\tau(t) = B(\tau) + A(t - \tau)$
  - hence,

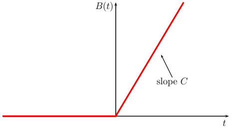
$$(B * A)(t) = \inf_{\tau \in \mathbb{R}} f_\tau(t)$$

- example:





$$A(t) = (\sigma + \rho t) I_{\{t \geq 0\}}$$



$$B(t) = Ct^+$$

  - in this example, $(B * A)(t) = \min\{A(t), B(t)\}$
  - is it always exactly the minimum?

Some Properties and Special Elements

- Additional properties of the $*$ operator
  - monotone: if $B \leq C$ then $A * B \leq A * C$
  - commutative: $A * B = B * A$
  - associative: $(A * B) * C = A * (B * C)$
  - distributive over min: $A * \min\{B, C\} = \min\{(A * B), (A * C)\}$
- Define

$$\delta(t) = \begin{cases} 0 & t < 0 \\ \infty & t \geq 0 \end{cases}$$

  - unit element:

$$\begin{aligned} (A * \delta)(t) &= \inf_{\tau \in \mathbb{R}}(A(\tau) + \delta(t - \tau)) \\ &= \inf_{t < \tau} A(\tau) = A(t) \end{aligned}$$

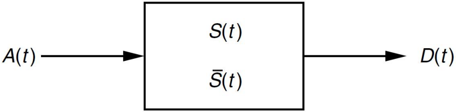
  - if B is causal then $B \leq \delta$: $A * B \leq A * \delta = A$
    - if A, B are causal then $A * B \leq \min\{A, B\}$
    - if A is causal then $A * A \leq A$
- Define

$$\delta_d(t) = \delta(t - d)$$

  - delay element: $(A * \delta_d)(t) = A(t - d) \stackrel{?}{=} \min\{A, \delta_d\}$

Overview and Definition

- consider an arrival sequence serviced by a network element



  - network element: dedicated link, link with cross traffic, path...
  - single dedicated link: removes $\leq Ct$ bits in interval of length t
- main focus: analyze the departure process!!

Definition. (Service Curves)

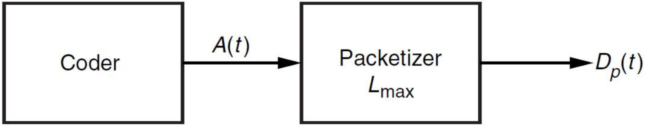
Let $S(t), \overline{S}(t)$ be non-negative, nondecreasing, causal functions

- if $A * S \leq D$, S is said to be a *lower service curve*
- if $A * \overline{S} \geq D$, $\overline{S}$ is said to be an *upper service curve*
- if $A * S = D$, S is said to be the *service curve*

Examples

- Example 1: voice packetizer
  - $A(t)$: encoder generates bursts at byte rate r (during burst)
  - packetizer collects $L_{\max}$ bytes, and sends packets
  - byte delay $\leq \frac{L_{\max}}{r}$
    - time to accumulate $L_{\max}$ bytes arriving at rate r
  - packetized output $D_p(t) \geq A(t - \frac{L_{\max}}{r})$

$\Rightarrow \delta_{\frac{L_{\max}}{r}}$ is a lower service curve for $D_p(t)$



Examples

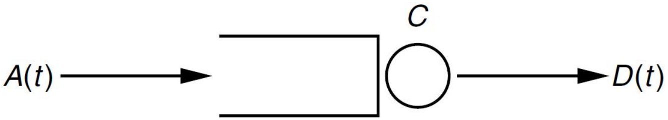
- Example 2: constant-rate server (infinite buffer)
  - fluid model: we have seen $D(t) = \inf_{s \in \mathbb{R}}(A(s) + C(t-s)^+)$

$\Rightarrow S(t) = Ct^+$ is *the* service curve

  - packetized model: a departure is considered at packet boundaries (size $L_{\max}$)

$\Rightarrow$ a lower service curve:

$$\begin{aligned}(Ct^+ - L_{max})^+ &= C(t - \tfrac{L_{\max}}{C})^+ \\ &= Ct^+ * \delta_{\frac{L_{\max}}{C}}\end{aligned}$$



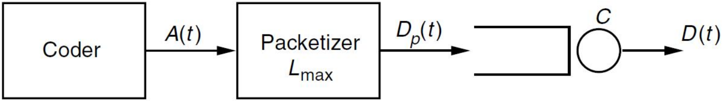
Examples

- Example 3: voice packetizer and constant-rate server
  - traffic entering the server is already packetized
  - no need to "pay" the delay twice
  - hence,

$$\begin{aligned} D &\geq D_p * Ct^+ \\ &= (A * \delta_{\frac{L_{\max}}{r}}) * Ct^+ \\ &= A * (\delta_{\frac{L_{\max}}{r}} * Ct^+) \end{aligned}$$

$\Rightarrow$ a lower service curve:

$$\delta_{\frac{L_{\max}}{r}} * Ct^+$$



Examples

- Example 3: voice packetizer and constant-rate server
  - traffic entering the server is already packetized
  - no need to "pay" the delay twice
  - hence,

$$\begin{aligned} D &\geq D_p * Ct^+ \\ &= (A * \delta_{\frac{L_{max}}{r}}) * Ct^+ \\ &= A * (\delta_{\frac{L_{max}}{r}} * Ct^+) \end{aligned}$$

$\Rightarrow$ a lower service curve:

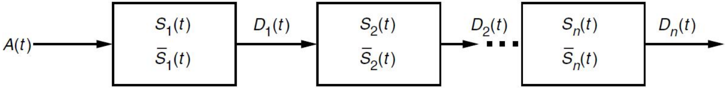
$$\delta_{\frac{L_{max}}{r}} * Ct^+$$

Defintion. (latency rate server)

A network element is a *latency rate server* with rate $r \in \mathbb{R}^+$ and delay $d \in \mathbb{R}^+$ if it has a lower service curve of the form

$$\delta_d * rt^+ = r(t-d)^+$$

Service Elements in Tandem



- concatenating service curves (e.g., lower service curves)

$$\begin{aligned} D_n &\geq D_{n-1} * S_n \\ &\geq (D_{n-2} * S_{n-1}) * S_n \\ &\geq \ldots \\ &\geq A * (S_1 * S_2 * \ldots * S_n) \end{aligned}$$

- example – rate latency servers: $S_i(t) = r_i(t - d_i)^+$

$$\begin{aligned} (S_1 * \ldots * S_n) &= (r_1(t - d_1)^+) * \ldots * (r_n(t - d_n)^+) \\ &= (r_1 t^+ * \delta_{d_1}) * \ldots * (r_n t^+ * \delta_{d_n}) \\ &= (r_1 t^+ * \ldots * r_n t^+) * (\delta_{d_1} * \ldots * \delta_{d_n}) \\ &= r t^+ * \delta_d = r(t - d)^+ \end{aligned}$$

where $r = \min_i r_i$ and $d = \sum_i d_i$

Obtaining a Better Departure Analysis

- recall: service curves aim at analyzing the departure process
- if we only know arrival's peak rate
  - results in over-provisioning
  - provides loose bounds
- goal: characterize *arrival* process better

Definition. (Envelope)

Givan any function $E(t)$, arrival process $A(t)$ has *envelope* E if for all t, and all $\tau \leq t$,

$$A(t) - A(\tau) \leq E(t - \tau)$$

- envelope: invariant to "when", depends only on "how long"
- example:
  - constant rate arrival: $A(t) = Rt^+$
  - A has envelope $E = A$:
    $A(t) - A(\tau) = Rt^+ - R\tau^+ \leq R(t - \tau)^+ = E(t - \tau)$

From Envelopes to Regulators

- A has envelope E *iff* $A \leq A * E$
- if A has a causal envelope E then $A = A * E$
  - $A * E \leq A * \delta = A$
- A process E is *sub-additive* if for all $\tau \leq t$

$$E(t) \leq E(\tau) + E(t - \tau)$$

  - if E is sub-additive, then $E \leq E * E$
  - example – leaky bucket: $E(t) = \sigma + \rho t$

$$E(t) = \sigma + \rho t \leq 2\sigma + \rho t = E(\tau) + E(t - \tau)$$

Definition. (Regulator)

A network element is a *regulator* with envelope E if for *any* input process A, the departure process D has envelope E. I.e.,

$$\forall t, \forall \tau \leq t \qquad D(t) - D(\tau) \leq E(t - \tau)$$

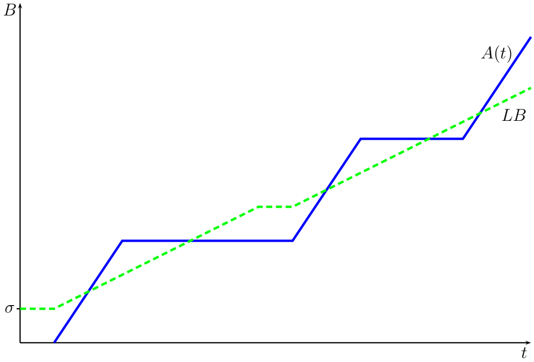
- equivalently, $D \leq D * E$

Leaky Bucket Example

Theorem.

For any arrival process $A(t)$ regulated by a leaky-bucket causal envelope $E(t) = \sigma + \rho t$, the departure process $D(t)$ satisfies

$$D = A * E$$

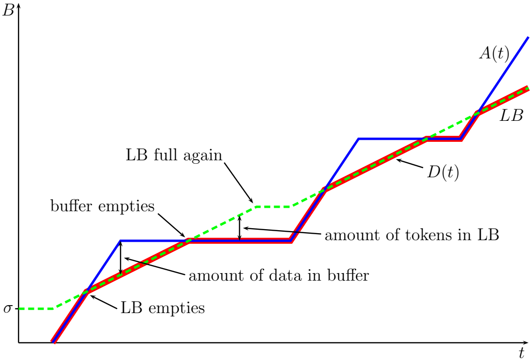


Leaky Bucket Example

Theorem.

For any arrival process $A(t)$ regulated by a leaky-bucket causal envelope $E(t) = \sigma + \rho t$, the departure process $D(t)$ satisfies

$$D = A * E$$



Leaky Bucket Example

Theorem.

For any arrival process $A(t)$ regulated by a leaky-bucket causal envelope $E(t) = \sigma + \rho t$, the departure process $D(t)$ satisfies

$$D = A * E$$

Proof.

- for any t, and $\tau \leq t$

$$\begin{aligned} D(t) &= D(\tau) + D(t) - D(\tau) \\ &\leq D(\tau) + E(t-\tau) \\ &\leq A(\tau) + E(t-\tau) \end{aligned}$$

$\Rightarrow D \leq A * E$

- it remains to show the converse, i.e., that $D \geq A * E$
- consider any time t
  - if the data buffer is empty at t, then we have

$$D(t) = A(t) = (A * \delta)(t) \geq (A * E)(t)$$

Leaky Bucket Example

Theorem.

For any arrival process $A(t)$ regulated by a leaky-bucket causal envelope $E(t) = \sigma + \rho t$, the departure process $D(t)$ satisfies

$$D = A * E$$

Proof.

- assume the data buffer is not empty at t
  - the LB *bucket* must be empty at t
- let $v = \sup\{s \leq t \mid \text{ the LB bucket is full at } s\}$ (i.e., v is the "start" of a busy period)

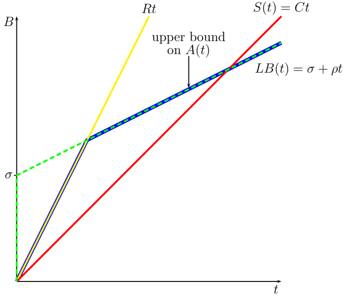
  $\Rightarrow$ until v_- all arrivals have passed through
- during $[v, t]$, the LB constantly uses up tokens, hence

$$\begin{aligned} D(t) &= D(v_-) + \sigma + \rho(t - v) \\ &= A(v_-) + E(t - v)) \\ &\geq \inf_\tau (A(\tau) + E(t - \tau)) = (A * E)(t) \end{aligned}$$

□

A New Dog Doing Old Tricks

- recall IntServ
  - single dedicated link with rate C
  - arrival process:
    - peak rate R
    - regulated by a (σ, ρ) LB envelope

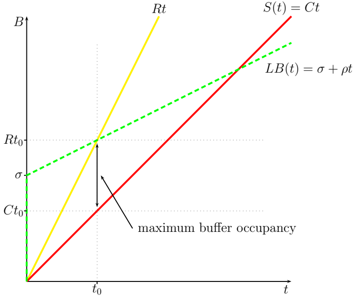


A New Dog Doing Old Tricks

- recall IntServ
  - single dedicated link with rate C
  - arrival process:
    - peak rate R
    - regulated by a (σ, ρ) LB envelope
- $R \cdot t_0 = \sigma + \rho \cdot t_0$

  $\Rightarrow \; t_0 = \frac{\sigma}{R-\rho}$
- maximum buffer occupancy:

  $$R \cdot t_0 - C \cdot t_0 = \frac{\sigma(R-C)}{R-\rho}$$



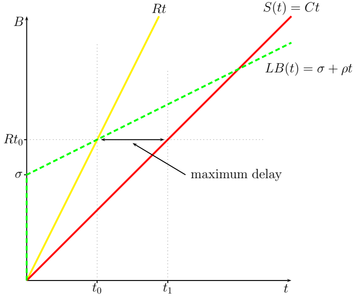
A New Dog Doing Old Tricks

- recall IntServ
  - single dedicated link with rate C
  - arrival process:
    - peak rate R
    - regulated by a (σ, ρ) LB envelope
- $R \cdot t_0 = \sigma + \rho \cdot t_0$

  $\Rightarrow \; t_0 = \frac{\sigma}{R-\rho}$
- maximum buffer occupancy:

  $$R \cdot t_0 - C \cdot t_0 = \frac{\sigma(R-C)}{R-\rho}$$
- $Ct_1 = Rt_0 = \frac{R\sigma}{R-\rho}$
- maximum delay:

  $$t_1 - t_0 = \frac{\sigma(R-C)}{C(R-\rho)}$$



A Network Calculus View

- Given:
  - arrival process $A(t)$ with envelope $E(t)$
  - lower service curve $S(t)$

  define:

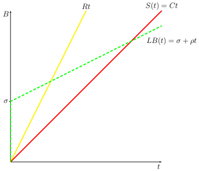
$$d_{\max} = \inf \{d \mid E * \delta_d \leq S\}$$

  - $d_{\max}$: least right shift of E s.t. it's below S, or
  - $d_{\max}$: least left shift of S s.t. it's above E
- For $S(t) = Ct^{+}$ it must hold for any t:

$$S(t + d_{\max}) = Ct + Cd_{\max} \geq E(t)$$

$\Rightarrow \; d_{\max} = \sup_{t \geq 0} \left(\frac{E(t) - Ct}{C} \right)$

- in the previous example:



$$\begin{aligned} E(t) &= \min(Rt^{+}, (\sigma + \rho t) I_{\{t \geq 0\}}) \\ &= (Rt^{+} * (\sigma + \rho t) I_{\{t \geq 0\}})(t) \end{aligned}$$

Delay and Provisioning

- $D \geq A * \delta_{d_{\max}}$
  - S is a lower service curve: $D \geq A * S$
  - by the definition of $d_{\max}$: $A * S \geq A * (E * \delta_{d_{\max}})$
  - associativity $+$ E is an envelope of A:

$$\begin{aligned} D &\geq (A * E) * \delta_{d_{\max}} \\ &\geq A * \delta_{d_{\max}} \end{aligned}$$

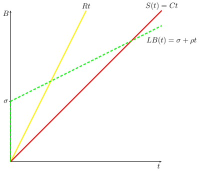
- Rate Provisioning for $S(t) = Ct^{+}$
  - assume we want the delay to be at most T:

$$d_{\max} = \sup_{t \geq 0} \left(\frac{E(t) - Ct}{C} \right) \leq T$$

  - any C satisfying this: $E(t) - Ct \leq CT$

$$\Rightarrow \qquad C \geq \sup_{t \geq 0} \left(\frac{E(t)}{T + t} \right)$$

  - $C_{\min}$: minimal rate satisfying this requirement
    - bound is tight



Generalizing Reich's Equation

Theorem. (Backlog bound)

If arrival A has envelope E, and S is a lower service curve on departures D, then the backlog at any time t satisfies

$$A(t) - D(t) \leq \sup_{\tau \geq 0} \{E(\tau) - S(\tau)\}$$

Proof.

By the definition of lower service curve S and envelope E

$$\begin{aligned} A(t) - D(t) &\leq A(t) - (A * S)(t) \\ &= A(t) - \inf_{0 \leq \tau \leq t}(A(t-\tau) + S(\tau)) \\ &= \sup_{0 \leq \tau \leq t}(A(t) - A(t-\tau) - S(\tau)) \\ &\leq \sup_{0 \leq \tau \leq t}(E(\tau) - S(\tau)) \end{aligned}$$

□

- i.e., the (max) vertical distance between E and S

A "Delay" Version of Reich's Equation

Theorem. (Delay bound)

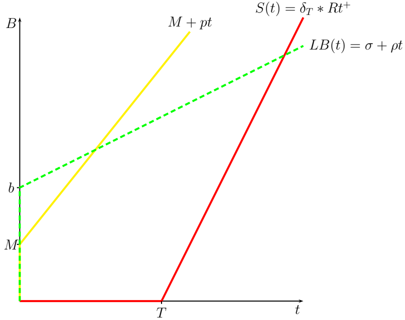
If arrival A has envelope E, and S is a lower service curve on departures D, then the delay $d(t)$ at any time t satisfies

$$d(t) \leq d_{\text{max}} = \inf \{ d \mid E * \delta_d \leq S \}$$

- i.e., the (max) horizontal distance between E and S

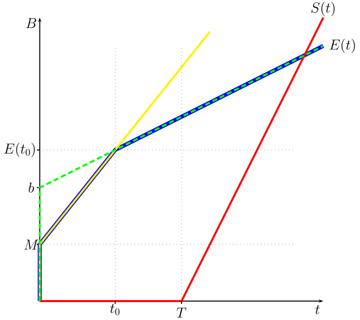
IntServ: Analyzing a (Path of) Latency-Rate Server(s)

- Recall IntServ TSpec: (M, p, r, b)
  - envelope $E(t) = (M + pt) * (b + rt) = \min\{M + pt, b + rt\}$
  - lower service curve $S(t) = \delta_T * Rt^+$



IntServ: Analyzing a (Path of) Latency-Rate Server(s)

- Recall IntServ TSpec: (M, p, r, b)
  - envelope $E(t) = (M + pt) * (b + rt) = \min\{M + pt, b + rt\}$
  - lower service curve $S(t) = \delta_T * Rt^+$
- Backlog and delay are obtained at vertices
  - Backlog $\leq \max\{E(T), E(t_0) - S(t_0)\}$
  - Delay $\leq \max\left\{\frac{E(t_0)}{R} + T - t_0, \frac{M}{R} + T\right\}$



IntServ: Analyzing a (Path of) Latency-Rate Server(s)

- Recall IntServ TSpec: (M, p, r, b)
  - envelope $E(t) = (M + pt) * (b + rt) = \min\{M + pt, b + rt\}$
  - lower service curve $S(t) = \delta_T * Rt^+$
- Backlog and delay are obtained at vertices
  - Backlog $\leq \max\{E(T), E(t_0) - S(t_0)\}$
  - Delay $\leq \max\left\{\frac{E(t_0)}{R} + T - t_0, \frac{M}{R} + T\right\}$
- E.g., delay bound translates to

$$\frac{M + \frac{b-M}{p-r}(p-R)^+}{R} + T$$

  - familiar???
- Recall:
  - concatentaing latency-rate servers $\equiv$ latency-rate server
  - effectively on a *path*!!
  - latency T = sum of latencies
  - rate R = minimum rate

Extensions and Current Research

- Stochastic Network Calculus
  - stochastic arrival curves, envelopes, algebra (min-sum operator), etc.
- Broad field (and thriving):
  - WFQ can be interpreted and analyzed within this framework
    - we'll see it later "stand-alone"
  - also related to Adversarial Queueing Theory
    - we'll see it later "stand-alone"
  - several Infocom papers every year
- Bibliography and recommended reading:
  - Kumar, Manjunath, and Kuri, "*Communication Networking: An Analytical Approach*", Elsevier, 2004.
  - Le Boudec and Thiran, "*Network Calculus: A Theory of Deterministic Queuing Systems for the Internet*", Springer, 2004.
  - many more...