Introduction to Network Calculus

John C.S. Lui

Department of Computer Science & Engineering
The Chinese University of Hong Kong
Outline

1. Introduction
2. The $(\sigma, \rho)$ constraints
3. $f$—upper constrained processes
4. Service Curve
What is network calculus?

- A theoretical framework to analyze performance guarantees (e.g., maximum delays, maximum buffer space requirements) in computer network.
- As traffic flows through a network, it is subject to constraints such as:
  - link capacity;
  - traffic shapers (e.g., leaky buckets);
  - congestion control;
  - background traffic.
- Express arrival, service and these constraints in a systematic manner (network calculus);
- key idea is to use the min-plus algebra.
Various Sections

- The basic \((\sigma, \rho)\) constraints, performance bounds of single queue.
- General constraint of deterministic constraint.
- Application to service curves.
- Given the input traffic and service curves, how to derive maximum delay for any server that conforms to the service curve.
- Applications: (a) bounding the maximum delay of a priority queue; (b) scheduling service at a constant rate link with multiple input streams in order to achieve a specified service curve.
Upper Constrained Arrival Process

- For simplicity, assume equal length packets transmitted in discrete time.
- A *cumulative arrival process* $A$ is a nondecreasing, integer-valued function on the nonnegative integer $\mathbb{Z}_+$ such that $A(0) = 0$.
- $A(t)$ denotes the number of arrivals in slots 1, 2, . . . , $t$.
- $a(t)$ is the number of arrivals at time $t$ and $a(t) = A(t) - A(t - 1)$.
- We said $A$ is *$(\sigma, \rho)$-upper constrained* (or $A \sim (\sigma, \rho)$) if
  \[ A(t) - A(s) \leq \sigma + \rho(t - s), \quad 0 \leq s \leq t. \]
- In this lecture, $\sigma, \rho$ are taken to be integer valued.
Example of $A$ and $\sigma$, $\rho$
Token Bucket Filter

- Token bucket filter, a popular way to regular data streams and to generate $(\sigma, \rho)$—upper constrained traffic.

**Definition**

A token bucket filter (with no dropping) with $(\sigma, \rho)$ operates like:

- The filter has infinite queue length and a token bucket.
- Events occur at integer time. New packets are added to the queue, and $\rho$ new tokens are added to the token bucket.
- As many packets immediately depart if each packet has a token.
- If there are more than $\sigma$ tokens in the bucket, drop some tokens until we have only $\sigma$ tokens.
Observation

A token bucket filter with parameter \((\sigma, \rho)\) is a \((\sigma, \rho)\)-regulator. Or for any input process \(A\), the output process \(B\) is \((\sigma, \rho)\)-upper constrained.

- Since at most \(\sigma\) tokens are in the bucket just before \(s + 1\).
- And \(\rho(t - s)\) tokens arrive in slots \(s + 1, \ldots, t\).
- At most \(\sigma + \rho(t - s)\) packets can depart from the filter in those slots.
- \(B\) is indeed upper constrained by \((\sigma, \rho)\).

Multiplexing Rule

If constrained flows are merged, the output process is also constrained, or

\[
A_i \sim (\sigma_i, \rho_i) \rightarrow \sum A_i \sim \left(\sum \sigma_i, \sum \rho_i\right)
\]
Performance bounds of constant server under $A \sim (\sigma, \rho)$

What are the performance bounds, i.e., duration of busy period, packet delay, for a constant server under $A \sim (\sigma, \rho)$?

- A single server with a constant service rate of $C$ (positive integer).
- Let $A$ be the cumulative arrival process.
- Let $q(t)$ be the queue length at time slot $t$. We have:

$$q(t + 1) = (q(t) + a(t + 1) - C)^+$$

with $q(0) = 0$. 
The \((\sigma, \rho)\) constraints

**Performance bounds: continue**

- Using induction on \(t\), we have

  \[
  q(t) = \max_{0 \leq s \leq t} \{ A(t) - A(s) - C(t - s) \} \tag{1}
  \]

  Show by induction: first, \(q(0) = 0\).

  \[
  q(1) = \max(0, q(0) + a(1) - C) = \max_{0 \leq s \leq 1} (A(1) - A(s) - C(1 - s)).
  \]

  Suppose it holds for \(t\), it follows

  \[
  q(t + 1) = \max\{0, \max_{0 \leq s \leq t} \{ A(t) - A(s) - C(t - s) \} + a(t + 1) - C\}
  = \max\{0, \max_{0 \leq s \leq t} \{ A(t + 1) - A(s) - C(t + 1 - s) \}\}
  = \max_{0 \leq s \leq t + 1} \{ A(t + 1) - A(s) - C(t + 1 - s) \}.
  \]
The output cumulative process $B$ satisfies:

$$B(t) = A(t) - q(t) = \min_{0 \leq s \leq t} \{A(s) + C(t - s)\} \quad \forall t \geq 0.$$

Queue Length Bound: Suppose $A$ is $(\sigma, \rho)$-upper constrained, if $C \geq \rho$, Eq (1) implies $q(t) \leq \sigma$ for all $t$. *(implication: We obtain the bound, independent of the service order)*

Conversely, if $C = \rho$ and $q(t) \leq \sigma$ for all $t$, then $A \sim (\sigma, \rho)$. *(implication: if we can control $q(t)$, we specify the envelop of $A$)*
Performance bounds: continue

We want to derive upper bound of

- busy period;
- packet delay

when \( A \sim (\sigma, \rho) \).

**Definition (Busy Period)**

Given time \( s \) and \( t \) with \( s \leq t \), a busy period is said to begin at \( s \) and end at \( t \) if \( q(s - 1) = 0, a(s) > 0, q(r) > 0 \) for \( s \leq r < t \) and \( q(t) = 0 \). The duration \( B \) of the busy period to be \( B = t - s \) time units.
The $(\sigma, \rho)$ constraints

Performance bound: continue

Observation

Given such busy period, we must have

- $C$ departures at each of the $B$ times $\{s, \ldots, t - 1\}$.
- At least one packet in the queue at time $t - 1$.
- At least $CB + 1$ packets must arrive at times $\{s, \ldots, t - 1\}$ to sustain the busy period.

Since $A \sim (\sigma, \rho)$, we have at most $\sigma + \rho B$ packets arrive in $B$. We have $CB + 1 \leq \sigma + \rho B$, we have $B \leq \frac{\sigma - 1}{C - \rho}$. If $B$ is an integer, it must be

$$B \leq \left\lfloor \frac{\sigma - 1}{C - \rho} \right\rfloor.$$
Performance bound: continue

Delay Bound

- The delay of a packet is the time the packet departs minus the time it arrives.
- The delay of any packet is less than or equal to the length of the busy period.

Thus, the upper of the packet delay $d$, independent of service discipline, is:

$$d \leq \left\lfloor \frac{\sigma - 1}{C - \rho} \right\rfloor.$$ 

If one unit of service time is added, we have

$$d + 1 \leq \left\lfloor \frac{\sigma - 1}{C - \rho} \right\rfloor \leq \left\lceil \frac{\sigma}{C - \rho} \right\rceil$$
Performance bound: continue

What if the service discipline is FIFO?

- If the packet has a nonzero waiting time, then it is carried over from the time it first arrived to the next time slot.
- The total number of packets carried over, including this packet, is less than or equal to $\sigma$ (shown after Eq. (1)).
- The delay of FIFO is:
  \[ d_{FIFO} = \left\lceil \frac{\sigma}{C} \right\rceil. \]
- If service time is included, we have to add 1 to the above expression.
Output Analysis 1

If \( A \sim (\sigma, \rho) \) and delay bound \( d \), what about the output \( B \)?

- Let say we know the maximum delay of the queue is \( d \).
- For \( s < t \), any packets that departs from the queue at a time in \( \{s+1, \ldots, t\} \) must arrive at one of the \( t - s + d \) times in \( \{s + 1 - d, \ldots, t\} \).
- Therefore, output process based on delay bound \( d \) is

\[
B(t) - B(s) \leq A(t) - A(s - d) \leq \sigma + \rho d + \rho(t - s).
\]

- Therefore, \( B \sim (\sigma + \rho d, \rho) \)-upper constrained.
If $A \sim (\sigma, \rho)$ and queue length bound $q$, what about the output $B$?

- Let say we know the maximum queue length is $q$.
- Let $q(t)$ be the queue length at time $t$, we have
  
  $$B(t) - B(s) = A(t) - A(s) - (q(t) - q(s))$$
  $$\leq A(t) - A(s) + q(s) \leq \sigma + \rho(t - s) + q$$
  $$\leq \sigma + q + \rho(t - s)$$

Therefore, output process based on queue length bound $q$ is $B \sim (\sigma + q, \rho)$-upper constrained.

Now we characterized $B$, $B$ is fed into another queue and we can continue to do the delay analysis.
Output analysis 3

If $A \sim (\sigma, \rho)$ and the server is work conserving, what about the output $B$?

- Assume it is a work conserving link with capacity $C$, we have

$$B(s) = \min_{0 \leq \tau \leq s} [A(\tau) + C(s - \tau)] \quad ; \quad B(t) = \min_{0 \leq \tau \leq t} [A(\tau) + C(t - \tau)]$$

- Let $\tau^*$ be the argument which achieves the minimum in $B(s)$. We have $B(s) = A(\tau^*) + C(s - \tau^*)$ and $B(t) \leq A(t - s + \tau^*) + C(s - \tau^*)$ (by choosing $\tau = t - s + \tau^*$). We have

$$B(t) - B(s) \leq A(t - s + \tau^*) + C(s - \tau^*) - A(\tau^*) - C(s - \tau^*)$$

$$= A(t - s + \tau^*) - A(\tau^*) \leq \sigma + \rho(t - s)$$

- Output process based on work conservation, $B$ is $(\sigma, \rho)$—upper constrained
Routing

Definition

An ideal router is a network element with one input $A$, one control input $P$, one output $B$ such that $B = P(A(t))$ where $A(t)$ is the cumulative number of arrival by time $t$, $P(n)$ is the number of arrivals that are selected among the first $n$ arrivals, and $B(t)$ as the cumulative departures by time $t$. In other words, the cumulative number of output by time $t$ is the cumulative number of arrivals selected by time $t$. 

$A(t)$ $\downarrow$ $P(n)$ $\downarrow$ $B(t)$

Router
Characterization of router’s output

Lemma

For an ideal router, if $A \sim (\sigma, \rho)$—upper constrained and $P \sim (\delta, \gamma)$—upper constrained, then $B \sim (\gamma \sigma + \delta, \gamma \rho)$—upper constrained.

Proof

Observe that

\[
B(t) - B(s) = P(A(t)) - P(A(s)) \
\leq \delta + \gamma (A(t) - A(s)) \
\leq \delta + \gamma (\sigma + \rho (t - s)) \
= \delta + \gamma \sigma + \gamma \rho (t - s)
\]
The \((\sigma, \rho)\) constraints

Application: feed-forward network

\[ A_1 \sim (\sigma_1, \rho_1) \]
\[ A_2 \sim (\sigma_2, \rho_2) \]
\[ A_3 \sim (\sigma_3, \rho_3) \]

\[ C_1 \]
\[ C_2 \]
\[ C_3 \]

Parameters

- \(C_1 = C_2 = C_3 = 4\)
- Arrival processes are \((\sigma_k, \rho_k)\) upper constrained with \((\sigma_1, \rho_1) = (1, 2), (\sigma_2, \rho_2) = (2, 1), (\sigma_3, \rho_3) = (3, 2)\).
- Routing, as indicated in the figure.
Analysis on the 1st communication link

Based on the multiplexing rule, the overall arrival to link 1 is 
\((\sigma_1 + \sigma_2, \rho_1 + \rho_2)\)–upper constrained, or \((3, 3)\).

Since \(\rho_1 + \rho_2 = 3 < C_1 = 4\), using the delay bound result after Eq. (1), the maximum queue length \(q_1\) in the first link is upper 
bounded by \(q_1 = \sigma_1 + \sigma_2 = 3\), and using the delay bound result, 
we have \(d_1 = \lceil (\sigma_1 + \sigma_2)/(C_1 - \rho_1 - \rho_2) \rceil = 3\).

Let \(B_1\) be the output process. Since \(A_2\) will not affect the second 
link, we only need to consider \(A_1\). Using the *bounding output process based on queue length*, we have 
\(B_1 \sim (\sigma_1 + q_1, \rho_1) = (4, 2)\)–upper constrained.
The ($\sigma, \rho$) constraints

Analysis on the 2nd communication link

- Based on the multiplexing rule, since $B_1 \sim (4, 2)$ and $A_3 \sim (3, 2)$, we have $A \sim (7, 4)$—upper constrained.
- Because 4 is equal to $C_2$, the maximum queue length $q_2 = 7$.
- Since $C_2 = 4$, we cannot use the delay bound result (since $C - \rho = 0$ in the denominator). If this link uses FCFS discipline, we have $d_2 = \lceil 7/4 \rceil = 2$.
- The output process $B_2$, based on bounding output process based on work conserving link is $B_2 \sim (7, 4)$—upper constrained.
Analysis on the 3rd communication link

- Arrival process to this link is same as $B_2$, therefore $(7, 4)$–upper constrained.
- Based on the established theory, bound on $q_3 = \sigma = 7$.
- Note that this bound is too loose. Why?
- Since $C_2 = C_3 = 4$, it means at most 4 packets come out from the 2nd link and these packets will be *immediately* served at the 3rd link.

Interesting questions

- Can we refine the theory to obtain tighter bound?
- What about communication systems with feedback?
- What about multi-class communication networks?
The $(\sigma, \rho)$ constraints

Single class non-feed-forward routing

**Definition**

For any increasing sequence $A$, we define its "stopped sequence" at time $\tau$, denoted as $A^{\tau}$, by

$$A^{\tau}(t) = \begin{cases} A(t) & \text{if } t \leq \tau, \\ A(\tau) & \text{otherwise} \end{cases} \quad (2)$$

**Comment**

If $A$ is an arrival process, then there are no further arrivals after time $\tau$ for the stopped sequence $A^{\tau}$. 
The $(\sigma, \rho)$ constraints

Traffic characterization of $A^\tau$

Lemma

For every $\rho$, a stopped sequence $A^\tau$ is $(\sigma(\tau), \rho)$–upper constrained where

$$\sigma(\tau) = \max_{0 \leq t \leq \tau} \max_{0 \leq s \leq t} [A(t) - A(s) - \rho (t - s)].$$

(3)

Proof

As the sequence $A^\tau$ is stopped at time $\tau$, $\sigma(\tau)$ is the maximum queue length of a work conserving link with capacity $\rho$ and input $A^\tau$. 
Traffic characterization of $A^\tau$: continue

**Corollary**

*If* $A^\tau$ *is* $(\sigma, \rho)$—upper constrained, *then* $\sigma(\tau) \leq \sigma$, *where* $\sigma(\tau)$ *is defined in Eq. (3).*

**Proof**

*If* $A^\tau$ *is* $(\sigma, \rho)$—upper constrained, *then* for all $0 \leq s \leq t \leq \tau$,

$$A(t) - A(s) = A^\tau(t) - A^\tau(s) \leq \sigma + \rho(t - s).$$

That $\sigma(\tau) \leq \sigma$ *follows immediately from Eq.(3).*
Example of feedback queues

Consider the following network:

- $A_1 \sim (\sigma_1, \rho_1)$
- $A_2 \sim (\sigma_2, \rho_2)$
- $P_{12}(B_2(t))$
- $P_{21}(B_1(t))$

What is the performance of the system? Say the queue length bound?

- $A_1 \sim (\sigma_1, \rho_1)$, $A_2 \sim (\sigma_2, \rho_2)$, $P_{12} \sim (\delta_{12}, \gamma_{12})$, $P_{21} \sim (\delta_{21}, \gamma_{21})$. 
Analysis

- Let $\tilde{A}_1$ ($\tilde{A}_2$) be the overall arrival process of the first (second) link and $B_1$ ($B_2$) be the respective output process. We have

$$\tilde{A}_1(t) = A_1(t) + P_{21}(B_2(t)),$$

$$\tilde{A}_2(t) = A_2(t) + P_{12}(B_1(t)).$$

- The main idea to for the analysis is to derive the performance bounds for a finite time $\tau$ and show the bounds are independent of $\tau$.

- Let $B_1^\tau$ ($B_2^\tau$) be the stopped sequence of $B_1$ ($B_2$) at time $\tau$.

- It follows that for "any" $\alpha_1$, $B_1^\tau \sim (\sigma_1(\tau), \alpha_1)$.

$$\sigma_1(\tau) = \max_{0 \leq t \leq \tau} \max_{0 \leq s \leq t} [B_1(t) - B_1(s) - \alpha_1(t - s)].$$

- Similarly, for "any" $\alpha_2$, $B_2^\tau \sim (\sigma_2(\tau), \alpha_2)$.

$$\sigma_2(\tau) = \max_{0 \leq t \leq \tau} \max_{0 \leq s \leq t} [B_2(t) - B_2(s) - \alpha_2(t - s)].$$
Analysis: continue

- Solve $\alpha_1$ and $\alpha_2$: $\alpha_1 = \rho_1 + \gamma_{21}\alpha_2$; $\alpha_2 = \rho_2 + \gamma_{12}\alpha_1$.

- Assume $\gamma_{12}\gamma_{21} < 1$, we have:

  $$\alpha_1 = (1 - \gamma_{12}\gamma_{21})^{-1}(\rho_1 + \gamma_{21}\rho_2) ; \quad \alpha_2 = (1 - \gamma_{12}\gamma_{21})^{-1}(\rho_2 + \gamma_{12}\rho_1).$$

- Using the routing and multiplexing rules we discussed:

  $$\tilde{A}_1 \sim (\sigma_1 + \gamma_{21}\sigma_2(\tau) + \delta_{21}, \rho_1 + \gamma_{21}\alpha_2),$$

  $$B_1^\tau \sim (\sigma_1 + \gamma_{21}\sigma_2(\tau) + \delta_{21}, \rho_1 + \gamma_{21}\alpha_2)$$

- Since we solved $\alpha_1, \alpha_2$, we can say that $B_1^\tau$ is $(\sigma_1 + \gamma_{21}\sigma_2(\tau) + \delta_{21}, \alpha_1)$–upper constrained. It follows that

  $$\sigma_1(\tau) \leq \sigma_1 + \gamma_{21}\sigma_2(\tau) + \delta_{21}.$$ 

- Using similar argument, we can characterize $B_2^\tau$:

  $$\sigma_2(\tau) \leq \sigma_2 + \gamma_{12}\sigma_1(\tau) + \delta_{12}.$$
The $(\sigma, \rho)$ constraints

Analysis: continue

- Solving the above equations results in $\sigma_1(\tau) \leq \tilde{\sigma}_1$ and $\sigma_2(\tau) \leq \tilde{\sigma}_2$ where
  \[
  \tilde{\sigma}_1 = (1 - \gamma_{12} \gamma_{21})^{-1}(\sigma_1 + \gamma_{21} \sigma_2 + \gamma_{21} \delta_{12} + \delta_{21}),
  \]
  \[
  \tilde{\sigma}_2 = (1 - \gamma_{12} \gamma_{21})^{-1}(\sigma_2 + \gamma_{12} \sigma_1 + \gamma_{12} \delta_{21} + \delta_{12}).
  \]

  These bounds are independent of $\tau$!! So $B_1$ is $(\tilde{\sigma}_1, \alpha_1)$–upper constrained and $B_2$ is $(\tilde{\sigma}_2, \alpha_2)$–upper constrained.

- This in turn implies that $\tilde{A}_1$ is $(\sigma_1 + \gamma_{21} \tilde{\sigma}_2 + \delta_{21}, \alpha_1)$–upper constrained and $\tilde{A}_2$ is $(\sigma_2 + \gamma_{12} \tilde{\sigma}_1 + \delta_{12}, \alpha_2)$–upper constrained.

- Queue length in server 1 is bounded by $\sigma_1 + \gamma_{21} \tilde{\sigma}_2 + \delta_{21}$ if $\sigma_1 = (1 - \gamma_{12} \gamma_{21})^{-1}(\rho_1 + \gamma_{21} \rho_2) \leq C_1$,

- Queue length in server 2 is bounded by $\sigma_2 + \gamma_{12} \tilde{\sigma}_1 + \delta_{12}$ if $\sigma_2 = (1 - \gamma_{12} \gamma_{21})^{-1}(\rho_2 + \gamma_{12} \rho_1) \leq C_2$. 
Generalization of \((\sigma, \rho)\)

**f–upper constraint processes**

- Let \(f\) be a nondecreasing function from \(\mathbb{Z}_+\) to \(\mathbb{Z}_+\).
- An arrival process \(A\) is \(f\)–upper constrained if
  \[
  A(t) - A(s) \leq f(t - s) \quad \text{for all } s, t \text{ with } 0 \leq s \leq t
  \]
- Rearranging, \(A\) is \(f\)–upper constrained iff \(A(t) \leq A(s) + f(t - s)\) for \(0 \leq s \leq t\), or \(A \leq A \star f\), where \(f \star g\) is the function on \(\mathbb{Z}_+\) defined as
  \[
  (f \star g) = \min_{0 \leq s \leq t} g(s) + f(t - s).
  \]

Similar to "convolution", the min-plus algebra uses min instead of integration, + instead of multiplication. **ILLUSTRATE!**
Some Comments

Some functions can be reduced without changing the condition that an arrival process is $f$–upper constrained, e.g., $f(0) = 0$ because $A(t) - A(t) = 0$ anyway.

Suppose $A$ is $f$–upper constrained and $s, u \geq 0$, then $A(s + u) - A(s) \leq f(u)$ but a tighter bound may be implied. Let $n \geq 1$ and $u$ is represented as $u = u_1 + \cdots + u_n$ where $u_i \geq 1$ and integer, then

$$A(s + u) - A(s) = (A(s + u_1) - A(s)) + (A(s + u_1 + u_2) - A(s + u_1)) + \cdots + (A(s + u_1 + \cdots + u_n) - A(s + u_1 + \cdots + u_{n-1})) \leq f(u_1) + f(u_2) + \cdots + f(u_n).$$
Sub-additive Closure

So $A(t) - A(s) \leq f^*(t - s)$, where $f^*$ is the sub-additive closure of $f$, is defined by

$$f^*(u) = \begin{cases} 
0 & \text{if } u = 0 \\
\min \{f(u_1) + \cdots + f(u_n) : n \geq 1, u_i \geq 1, \sum_i u_i = u\} & \text{if } u \geq 1.
\end{cases}$$
Properties on $f^*$

1. $f^* \leq f$

2. $A$ is $f$–upper constrained iff $A$ is $f^*$–upper constrained

3. $f^*$ is sub-additive, $f^*(s + t) \leq f^*(s) + f^*(t)$ for all $s, t \geq 0$

4. If $g$ is any other function with $g(0) = 0$ satisfying (1) and (3), then $g \leq f^*$

Illustrate
Maximal regulator for $f$

**Definition (Regulator for $f$)**

A regulator for $f$ is a service center such that for any input $A$, the corresponding output $B$ is $f$–upper constrained.

**Definition (Maximal Regulator for $f$)**

A regulator is said to be a maximal regulator for $f$ if the following is true. For any input $A$, if $B$ is the output of the regulator for input $A$ and if $\tilde{B}$ is a cumulative arrival process such that $\tilde{B} \leq A$ and $\tilde{B}$ is $f$–upper constrained, then $\tilde{B} \leq B$. 
**Finding the maximal regulator for f**

**Theorem**

A maximal regulator for $f$ is determined by $B = A \star f^*$.  

**Proof**

Let $A$, $B$, and $\tilde{B}$ be as in the definition of the maximal regulator, then

$$\tilde{B} = \tilde{B} \star f^* \leq A \star f^* = B$$  \hspace{1cm} (7)

- First equality holds because $\tilde{B}$ is $f$–upper constrained.
- Inequality holds because $\star f^*$ is a monotone operation.
- The final equality holds by the definition of $B$. 
Corollary

Suppose $f_1$ and $f_2$ are nondecreasing functions on $\mathbb{Z}^+_+$ with $f_1(0) = f_2(0) = 0$. Two maximal regulators in series, consisting of a maximal regulator of $f_1$ followed by a maximal regulator for $f_2$, is a maximal regulator for $f_1 \star f_2$. In particular, the output is the same if the order of the two regulators is reversed.

Proof

Let $A$ be in the input to the first regulator and $B$ is the output of the second regulator, then

$$B = (A \star f_1^*) \star f_2^* = A \star (f_1^* \star f_2^*) = A \star (f_1 \star f_2)^*.$$  \hspace{1cm} (8)

The last equality depends on the assumption $f_1(0) = f_2(0) = 0$. The second part of the theorem is by the uniqueness of maximal regulators and the fact $f_1 \star f_2 = f_2 \star f_1$. 

Maximal regulator for token bucket filter

**Theorem**

The token bucket filter with parameter $(\sigma, \rho)$ is the maximal $(\sigma, \rho)$ regulator.
Introduction

What we have learnt

- So far, we have considered passing a constrained process into a maximal regulator.
- Examples of maximal regulators are token bucket, queue with fixed service rate.
- Output of these servers is completely determined by the input.
- In practice, there can be many other types of servers, and server may have priority in selecting with traffic to serve first.
- What we need is a flexible way to specify a guarantee that a particular server offers.
Most service centers are not fixed rate server or token bucket filter, we need a flexible way to specify a service behavior.

**Definition (Service Curve)**

A service curve is a nondecreasing function from $\mathbb{Z}_+$ to $\mathbb{Z}_+$. Given a service curve $f$, a server is an $f$–server if for any input $A$, the output $B$ satisfies $B \geq A \ast f$. That is:

$$B(t) \geq \min_{0 \leq s \leq t} \{A(s) + f(t - s)\}$$
comment on service curves under study

Comments

- For “regulator”, we can completely specify the output process $B$. For “service curve”, we get the “inequality” only.

- We only consider servers such that $B(t) \leq A(t)$ (or causality), and it is assumed that $A(0) = 0$. Take $s = t$ in the above equation implies $B(t) \geq A(t) + f(0)$ and this can only happen when $f(0) = 0$. 
Examples

Different servers

- Given an integer \( d \geq 0 \), define

\[
O_d(t) = \begin{cases} 
0 & \text{for } t \leq d, \\
+\infty & \text{for } t > d.
\end{cases}
\]

Then a FIFO device is an \( O_d \)-server iff the delay for every packets is less than or equal to \( d \), independent of \( A \).

- A server with constant service rate \( C \) is an \( f \) server for \( f(t) = Ct \).

- A leaky bucket regulator is an \( f \) server for \( f(t) = (\sigma + \rho t)I_{\{t\geq1\}} \).

- The maximal regulator for a function \( f \) is an \( f^* \)-server.
Suppose an $f_1$–upper constrained process $A$ passes through an $f_2$–server. We define:

**Definition ($d_V$)**

$$d_V = \max_{t \geq 0} (f_1^*(t) - f_2(t)),$$

or $d_V$ is the maximum vertical distance that the graph $f_1^*$ is above $f_2$.

**Definition ($d_H$)**

$$d_H = \max \{ d \geq 0 : f_1(t) \leq f_2(t + d) \text{ for all } t \geq 0 \},$$

or $d_H$ is the maximum horizontal distance that the graph $d_2$ is to the right of $f_1^*$.
Theorem

Let $A$ be $f_1$–upper constrained passing through an $f_2$–server with output process $B$. The queue size $A(t) - B(t)$ is less than or equal to $d_V$ for any $t \geq 0$, and if the order of service is FIFO, the delay of any packet is less than or equal to $d_H$. 
Proof

Let $t \geq 0$. Since it is an $f_2$--server, there exists an $s^*$ with $0 \leq s^* \leq t$ such that $A(t) \leq A(t - s^*) + f_1(s^*)$. Because $A$ is $f_1$--upper constrained, $A(t) \leq A(t - s^*) + f_1(s^*)$. Thus, $A(t) - B(t) \leq f_1(s^*) - f_2(s^*) \leq d_v$.

Suppose a packet arrives at time $t$ and departs at time $\bar{t} > t$. Then $A(t) > B(\bar{t} - 1)$. Since the server is $f_2$--server, there exists an $s^*$ with $0 \leq s^* \leq \bar{t} - 1$ such that $B(\bar{t} - 1) \geq A(s^*) + f_2(\bar{t} - 1 - s^*)$. Because $A(t) > B(\bar{t} - 1)$, it must be that $0 \leq s^* \leq t$. Since $A$ is $f_1$--upper constrained, we have:

$$A(s^*) + f_1(t - s^*) \geq A(t) \geq B(\bar{t} - 1) \geq A(s^*) + f_2(\bar{t} - 1 - s^*).$$

so that $f_1(t - s^*) > f_2(\bar{t} - 1 - s^*)$. Hence $\bar{t} - 1 - t < d_H$, so $\bar{t} - t < d_H$. 
Example

Consider a server of constant rate $C$ which servers input streams 1 and 2, giving priority to packets from input 1, and serves the packets within a single input stream in FIFO order. Let $A_i$ be the cumulative arrival stream of input $i$.

Claim

If $A_1$ is $f_1$–upper constrained, the link is an $\tilde{f}_2$–server for type 2 stream, where

$$\tilde{f}_2(t) = (Ct - f_1(t))^+.$$
Example

Suppose $A_i$ is $(\sigma_i, \rho_i)$ constrained for each $i$. Then
\[
\tilde{f}(t) = \left( (C - \rho_1) t - \sigma_1 \right)^+.\]
Applying previous result to yield the delay for any packet in input 2 is less than or equal to $(\sigma_1 + \sigma_2)/(C - \rho_1)$. Since packets in input 1 are not affected by the packets from input 2, the delay for any packet in input 1 is less than or equal to $\lceil \sigma_1/C \rceil$.

If queue were served in FIFO order, then the maximum delay for packets from either input is $\lceil \sigma_1/C \rceil$. If $\sigma_1$ is much smaller than $\sigma_2$, the delay for input 1 packets is much smaller for the priority server than for the FIFO server.
Suppose there are multiple input streams on the constant rate link to meet specified service curves for each input. Let the $i^{th}$ input has a cumulative arrival process $A_i$ which is known to be $g_i$—upper constrained and supposed that the $i^{th}$ input wants to receive service conforming to a specified service curve $f_i$. The question is, what is the algorithm to achieve the above goal?
Service Curve Earliest Deadline (SCED)

SCED

- Under SCED, for each input stream $i$ let $N_i(t) = A_i(t) \star f_i$. Then $N_i(t)$ is the minimum number of type $i$ packets that must depart by time $t$ in order that the input $i$ see service curve $f_i$. Based on this, the deadline can be computed for each packet of input $i$. Specifically, the deadline for the $k^{th}$ packet from input $i$ is the minimum $t$ such that $N_i(t) \geq k$. In other words, if all packets are scheduled by their deadlines, then all service curve constraints are met.

- Given any arrival sequence with deadlines, if it is possible for an algorithm to meet all the deadlines, then the **earliest deadline first (EDF)** scheduling can do it. The SCED is to put deadlines on the packets and use EDF.
Theorem

Given $g_i, f_i$ for each $i$ and capacity $C$ satisfying

$$\sum_{i=1}^{n} (g_i \star f_i)(t) \leq Ct \quad \forall t,$$

the service to each input stream $i$ provided by the SCED scheduling algorithm conforms to service curve $f_i$. 
Proof

- Fix a time $t_0 \geq 1$. Let say all packets with deadline $t_0$ or earlier are colored red, while all other packets are colored white. For any time $t \geq 0$, let $q_0(t)$ denote the number of red packets that are not scheduled by $t_0$. Since EDF is used, red packets have pure priority over the white packets, so $q_0$ is the queue length process in case all white packets are all ignored. We need to show $q_0(t_0) = 0$.

- For $0 \leq s \leq t_0 - 1$, the number of red packets that arrive from stream $i$ in the set of times $\{s + 1, \ldots, t_0\}$ is $(N_i(t_0) - A_i(s))^+$. Also, $N_i(t_0) \leq A_i(t_0)$. Therefore

$$q_0(t_0) = \max_{0 \leq s \leq t_0} \left\{ \sum_i (N_i(t_0) - A_i(s))^+ \right\} - C(t_0 - s).$$
Proof: continue

For any $s$ with $1 \leq s \leq t_0$,

$$N_i(t_0) = (A_i * f_i)(t_0) \leq (A_i * g_i * f_i)(t_0)$$

$$= \min_{0 \leq u \leq t_0} A_i(u) + (g_i * f_i)(t_0 - u) \leq A_i(s) + (g_i * f_i)(t_0 - s)$$

It follows that

$$\sum_i (N_i(t_0) - A_i(s))^+ \leq \sum_i (g_i * f_i)(t_0 - s) \leq C(t_0 - s).$$

So $q_0(t_0) = 0$ and the proposition is proved.
Consider the following network:

We have \( \rho_1 + \rho_2 \leq C_1 \) and \( \rho_1 + \rho_3 \leq C_2 \). The first server is FIFO server. The second server gives priority to \( A_3 \) but is FIFO in within each class. \((\sigma_i, \rho_i) = (4, 2)\) for \( 1 \leq i \leq 3 \) and \( C_1 = C_2 = 5 \).

(a) What is the maximum delay \( d_1 \)? (b) Characterize \( B_1 \). (c) What is the maximum delay of stream 1 in the second queue?
Solution:

- The total arrival stream to the first queue is 
  \((\sigma_1 + \sigma_2, \rho_1 + \rho_2)\)–upper constrained. Since the server is FIFO, 
  \(d_1 = \lceil (\sigma_1 + \sigma_2)/C_1 \rceil = 2\).

- \(B \sim (\sigma_1 + \rho_1 d_1, \rho_1)\)–upper constrained.

- Let \(d_1^2\), be the delay of the lower priority stream for a FIFO server is:

  \[d_1^2 \leq \frac{(\sigma_1 + \rho_1 d_1) + \sigma_3}{C_3 - \rho_3} = \frac{12}{3} = 4.\]