Network Calculus Tests – Naming Scheme

Version 1.1 (2014-Dec-30)



 \bigodot Steffen Bondorf 2013 - 2014, Some Rights Reserved.

Except where otherwise noted, this work is licensed under Creative Commons Attribution-Share Alike 4.0.

See http://creativecommons.org/licenses/by-sa/4.0/

General Information

The network calculus naming scheme presented in this document was created for the purpose of testing the Disco Deterministic Network Calculator $(DiscoDNC)^1$ – an open-source deterministic network calculus tool developed by the *Distributed Computer Systems (DISCO) Lab* at the University of Kaiserslautern.

Changelog:

Version 1.1 (2014-Dec-30): Match latest publications.

- Added definition of token-bucket arrival curves \mathcal{F}_{TB} and rate-latency \mathcal{F}_{RL} service curves to clear up the associated variables.
- Renamed the output bound quantifier from * to '.
- Renamed the newly merging cross-traffic function from x' to \bar{x} .
- Changed the explicit path quantifier from $s_n s_m$ to $\langle s_i, s_j \rangle$.
- Added the flow-based path quantifier $P(f_n)$.
- Added the operators used in the tests.
- Minor fixes in the examples.

¹http://disco.cs.uni-kl.de/index.php/projects/disco-dnc

Naming Scheme

$variable_{local_quantifier}^{semantic_quantifier}$

- variables:
 - $-\alpha$: arrival curve
 - β: service curve
 - $-\alpha_{r,b} \in \mathcal{F}_{TB}$ token-bucket arrival curves:

$$\mathcal{F}_{TB} = \{ \gamma_{r,b} \colon \mathbb{R}^+ \to \mathbb{R}^+ \mid \gamma_{r,b} (0) = 0, \forall d > 0 : \gamma_{r,b} (d) = b + r \cdot d \}, r, b \ge 0$$

 $-\beta_{R,T} \in \mathcal{F}_{\mathrm{RL}}$ rate latency service curves:

$$\mathcal{F}_{RL} = \left\{ \beta_{R,T} : \mathbb{R}^+ \to \mathbb{R}^+ \, | \, \beta_{R,T} (d) = R \cdot [d-T]^+ \right\}, \, T \ge 0, \, R > 0$$
with $R \cdot [d-T]^+ = [R \cdot (d-T)]^+ = \max\{0, R \cdot (d-T)\}$

- − b: burst
- T: latency
- -r, R: rate
- B: backlog bound
- − D: delay bound
- semantic quantifiers for arrival curves α and its variables (e.g., burst b):
 - f_n : arrival curve of flow f_n
 - $\{f_n, \ldots, f_m\}$: sum of arrival curves of flow f_n, \ldots, f_m
 - $-x(f_n)$: arrival curve of all crossflows of flow f_n (needs local quantification)
 - $-\bar{x}(f_n)$: arrival curve of newly joining crossflows of flow f_n (needs local quantification)
 - ': output bound (needs local quantification to id service curve)
 - no quantifier given: sum of all arrivals (needs local quantification)
- semantic quantifiers for service curves β and its variables (e.g., latency T):
 - l.o. f_n : left-over for flow f_n (needs local restriction)
 - * SFA l.o. f_n : SFA left-over for flow f_n (needs local restriction)
 - * PMOO l.o. f_n : PMOO left-over for flow f_n (needs local restriction)
 - no quantifier given: unaltered variable (needs local quantification)
- local quantifiers:
 - $-s_i$: at server s_i
 - $-\langle s_i, s_i \rangle$: on sub-path (see semantic quantifier) between s_i and s_j :
 - * α : data arrivals on link from s_i to s_j , i.e., there must be a direct link
 - * β : convolved service curve on the path from s_i to s_j (both included)
 - e2e: end-to-end (only in conjunction with β as well as it's rate R and latency T)
 - $-P(f_n)$: on the path of flow f_n

Operators

- $(\min, +)$ -convolution: \otimes
- (min, +)-deconvolution: \oslash
- $\bullet\,$ non-decreasing, non-negative subtraction: \ominus
- \bullet pointwise addition: +
- pointwise subtraction: —

Examples

$$f_0$$
 f_0
 f_1
 f_2
 f_1

•
$$\alpha^{f_0} = \alpha^{f_0}_{s_0} = \alpha^{x(f_3)}_{s_0}$$

•
$$\alpha_{s_1}^{f_2} = \alpha_{s_2s_1}^{f_2} = (\alpha_{s_2}^{f_2})' = (\alpha^{f_2})'$$

•
$$\alpha_{s_1}^{x(f_2)} = \alpha_{s_0s_1}^{f_0} = (\alpha_{s_0}^{f_0})' = (\alpha^{f_0})'$$

•
$$\alpha_{s_2}^{x(f_1)} = \alpha^{f_2+f_3} = \alpha^{f_2} + \alpha^{f_3}$$

•
$$\alpha_{s_2} = \alpha_{s_2}^{f_1} + \alpha_{s_2}^{f_2} + \alpha_{s_2}^{f_3} = \sum_{n=1}^{3} \alpha_{s_2}^{f_n} = \sum_{n=1}^{3} \alpha^{f_n} = \alpha_{s_2}^{f_1 + f_2} + \alpha^{f_3} = \alpha^{f_1 + x(f_1)}$$

•
$$\alpha_{s_2} = \alpha^{f_n} + \alpha_{s_2}^{x(f_n)}, n \in \{1, 2, 3\}$$

•
$$\alpha_{s_3} = \alpha_{\langle s_0, s_3 \rangle} + \alpha_{\langle s_1, s_3 \rangle} + \alpha_{\langle s_2, s_3 \rangle} = \alpha_{\langle s_0, s_3 \rangle}^{f_3} + \alpha_{\langle s_1, s_3 \rangle}^{f_0} + \alpha_{\langle s_2, s_3 \rangle}^{f_1} = (\alpha_{s_0}^{f_3})' + (\alpha_{s_1}^{f_0})' + (\alpha_{s_2}^{f_1})'$$

•
$$\alpha_{s_3}^{x(f_0)} = (\alpha_{s_0}^{x(f_0)})' + (\alpha_{s_0} - \alpha_{s_0}^{x(f_1)})' = ((\alpha^{f_3})')' + (\alpha^{f_1})'$$

$$\bullet \ \alpha_{s_3}^{x(f_3)} = \alpha_{s_3} - \alpha_{s_3}^{f_3} = \left(\alpha_{\langle s_0, s_3\rangle}^{f_3} + \alpha_{\langle s_1, s_3\rangle}^{f_0} + \alpha_{\langle s_2, s_3\rangle}^{f_1}\right) - \alpha_{\langle s_0, s_3\rangle}^{f_3} = \alpha_{\langle s_1, s_3\rangle}^{f_0} + \alpha_{\langle s_2, s_3\rangle}^{f_1}$$

$$\bullet \ \beta_{s_0}^{\mathrm{l.o.}f_0} = \left(\beta_{s_0} \ominus \alpha_{s_0}^{x(f_0)}\right) = \left(\beta_{s_0} \ominus \left(\alpha_{s_2}^{f_3}\right)'\right) = \left(\beta_{s_0} \ominus \left(\alpha_{s_2}^{f_3} \oslash \beta_{s_2}^{\mathrm{l.o.}f_3}\right)\right)$$

$$\bullet \ \beta_{\text{e2e}}^{\text{l.o.}f_0} = \beta_{s_0}^{\text{l.o.}f_0} \otimes \beta_{s_1}^{\text{l.o.}f_0} \otimes \beta_{s_3}^{\text{l.o.}f_0} = \left(\beta_{s_0} \ominus \alpha_{s_0}^{x(f_0)}\right) \otimes \left(\beta_{s_2} \ominus \alpha_{s_2}^{x(f_0)}\right) \otimes \left(\beta_{s_3} \ominus \alpha_{s_3}^{x(f_0)}\right)$$

•
$$\beta_{\text{e2e}}^{\text{l.o.}f_0} = \beta_{\langle s_0, s_3 \rangle}^{\text{l.o.}f_0} = \beta_{\langle s_0, s_1 \rangle}^{\text{l.o.}f_0} \otimes \beta_{s_3}^{\text{l.o.}f_0}$$

•
$$\beta_{\langle s_0, s_1 \rangle} = \beta_{s_0} \otimes \beta_{s_1} = \bigotimes_{i=0}^1 \beta_{s_i}$$

•
$$\beta_{P(f_3)}^{f_3} = \beta_{\langle s_2, s_3 \rangle}^{f_3} = \beta_{s_2} \otimes \beta_{s_0} \otimes \beta_{s_3} = \bigotimes_{i=\{2,0,3\}} \beta_{s_i}$$