Analysis and Evaluation of Per-Flow Delay Bound for Multiplexing Models

Yanchen Long
Zhejiang University, KTH Royal Institute of Technology

Zhonghai Lu
KTH Royal Institute of Technology

Xiaolang Yan
Zhejiang University

***Abstract*—Multiplexing models are common in resource sharing communication media such as buses, crossbars and networks. While sending packets over a multiplexing node, the packet delay bound can be computed using network calculus models. The tightness of such delay bound remains an open problem. This paper studies the multiplexing models for weighted round robin scheduling with different traffic arrival curves, and analyzes per-flow packet delay bounds with different service properties. We empirically evaluate the tightness of the delay bounds. Our results show the quality of different analysis models, and how influential each parameter is to tightness.**

I. Introduction

As modern CMPs and MPSoCs advance from multicore to many-core systems, on-chip interconnects play an increasingly important role in the system architectures. To satisfy the need of real-time applications, providing performance guarantees in the on-chip communication subsystem is indispensable in the entire system chain together with the computation and storage subsystems. In offering predictable performance, it is essential to build proper analytic models to analyze performance bounds under resource sharing scenarios.

In advanced on-chip interconnects, channel multiplexing is a most common resource sharing pattern. For instance, in a network-on-chip, routers are connected with channels to deliver packets through multiple hops from sources to destinations. We call a unicast stream of packets a flow. Along the flow's path, packets may experience contention with packets from other flows. Bounding the maximum packet delay is a critical requirement to ensure predictable communication performance. Surprisingly, for such a common pattern, the studies are not satisfactory in the sense that the tightness of multiplexing analysis models has not been sufficiently evaluated. This largely hinders our understanding on the basic resource sharing pattern in offering performance guarantees.

In the paper, we analyze the delay bound under Weighted Round Robin (WRR) multiplexing model based on Network Calculus (NC). We consider two different arrival models, specifically, the (b, r) model and TSPEC model (see Sec. III-A), and two different arbitration service properties, i.e. isolation property (see Sec. IV-B) and left-over service property (see Sec. IV-C). Moreover, we give a systematic evaluation on their tightness with respect to the characteristics of the service and flows (see Sec. V). Through our studies, we find out when and why one model is better than the other and in general under what traffic settings the tight delay bound may appear. It helps to shorten the simulation time and predict on which situation a worst case would happen.

978-3-9815370-2-4/DATE14/©2014 EDAA

II. Related Work

Network calculus is a theory of queuing systems based on highly abstract modeling of arrival curves and service curves. In [1] Cruz first described traffic as (b, r) characterization and gave a calculus method to obtain the delay. The concept of service curve was later formalized in [2]. Now, network calculus is also widely used in embedded systems, SoCs and other kind of networks besides Internet and ATM. On the other hand, simulation always acts as a powerful tool to validate different analysis models, though simulation is time-consuming and difficult due to the uncertainty of when the worst case would happen [3].

Chang discussed delay bound for aggregate scheduling in [4]. In [5], explicit results for multiplexing nodes with FIFO scheduling, Guaranteed Rate (GR) node, and strict service curve were provided. In [6], Qian et al. gave a specific method for analyzing the WRR arbitration.

III. Network Calculus Basics

A. Arrival curve: (b, r) and TSPEC models

Flow R has α as an arrival curve if and only if $\forall s < t : R(t) - R(s) \leq \alpha(t - s)$ [5]. Consider an affine arrival curve $\gamma_{r,b} = \alpha(t) = \begin{cases} rt + b, if & t > 0 \\ 0, otherwise \end{cases}$. It allows a traffic source to tolerate b bits instantaneously, but no more than $b + rt$ bits over any time interval t. This is the (b, r) model with burstiness b and rate r.

The (b, r) model well describes average traffic behavior but without consideration of the peak traffic behavior. To take both kinds of behavior into account, TSPEC (Traffic Specification) model [7] is proposed, and thus gives a better traffic characterization. It is a 4-element tuple (p, M, r, b) in the form $\alpha(t) = min(M + pt, rt + b)$ with maximum packet size M, peak rate p, burstiness b and sustainable rate r.

B. Service curve: Latency-rate server

Consider that a flow passes through system S with input and output function R and R^*. We say that S offers to the flow a service curve β if and only if β is wide-sense increasing, $\beta(0) = 0$ and $R^* \geq R \otimes \beta$, where $\otimes$ is the min-plus convolution, $f \otimes g(t) = inf_{0 \leq s \leq t}[f(s) + g(t - s)]$, and inf is "infimum" or "minimum" whenever applicable. If $\beta = R[t - T]^+ = \begin{cases} R(t - T), if & t > T \\ 0, otherwise \end{cases}$, β is called the latency-rate service curve with minimum rate R and maximum latency T [8], denoted by $\beta_{R,T}$.

C. Per-flow delay bound

Assume that a flow constrained by arrival curve α traverses a lossless system offering a service curve β. The delay bound $\bar{D}$ is the maximum horizontal deviation between α and β, $h(\alpha, \beta)$. Consider a flow constrained by $\gamma_{r,b}$ and served in a node with service curve $\beta_{R,T}$, the per-flow delay bound is

$$\bar{D}_{br} = T + \frac{b}{R}, r \leq R \ .$$

If a flow defined with TSPEC (p, M, r, b), is served in a node with service $\beta_{R,T}$, the maximum delay is bounded by

$$\bar{D}_{tsp} = \frac{M + \frac{b-M}{p-r}(p-R)^+}{R} + T, r \leq R \text{ [5].}$$

In general, the TSPEC model gives tighter analytic bound. When $p > R$, the larger R is than r, the larger difference exists between $\bar{D}_{tsp}$ and $\bar{D}_{br}$. See Fig. 1.

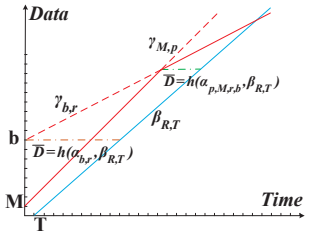

Fig. 1. (b, r) vs. TSPEC model

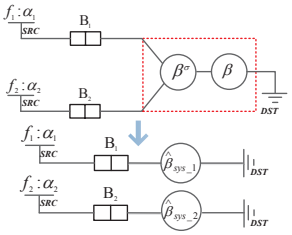

Fig. 2. Multiplexing analysis model

IV. Multiplexing Model and Analysis

A. The Multiplexing Model

Link sharing means that multiple flows from different buffers go through the same server, sharing the same outport and link bandwidth. As shown in Fig. 2, two flows have their own buffers accordingly.

Throughout this paper, we assume that there are two flows f_1 and f_2 sharing the same link, characterized as arrival curve α_1, α_2; FIFO B_i for flow f_i is large enough to avoid any packet loss or back pressure. (In fact, the buffer requirement can be calculated by Theorem 1.4.1 in [5].) f_1 is the tag flow. The arbiter provides a service curve $\beta^{\sigma}_{R_\sigma,T_\sigma}$. The sink provides service curve $\beta_{R,T}$. The tag flow f_1 gets the Equivalent Service Curve (ESC) $\hat{\beta}_{sys\_1} = \beta_{R_{sys},T_{sys}}$. $\bar{D}_{br}$ and $\bar{D}_{tsp}$ stand for delay bounds by the (b, r) model and TSPEC model, D_{sim} is the maximum observed delay, for the tag flow. We define their ratios $\xi_{br} = \frac{D_{sim}}{\bar{D}_{br}}$, $\xi_{tsp} = \frac{D_{sim}}{\bar{D}_{tsp}}$ as tightness, an evaluation criterion for the reachability of the calculus delay bound. In our analysis, we implement WRR arbitration, with each flow having a pre-set weight ϕ_i.

B. Analysis with Isolation Property

In the WRR arbitration, each flow is allocated a time slot equal to its weight ϕ_i. For each round, the arbiter ensures flow f_i at least ϕ_i packets to go in its time slot. The worst waiting time appears when flow f_i just misses its time slot in this round. Consider an arbiter serves R_σ packets/cycle, then the minimum rate guaranteed to flow f_i is $R_\sigma \frac{\phi_i}{\sum_{j=1}^{j=N} \phi_j}$, where N is the number of flows. The maximum waiting time flow f_i encounters in each round is $\frac{\sum_{j\neq i} \phi_j}{R_\sigma}$, described as a burst delay function $\delta_{\frac{\sum_{j\neq i} \phi_j}{R_\sigma}}(t) = \begin{cases} +\infty, if \quad t > \frac{\sum_{j\neq i} \phi_j}{R_\sigma} \\ 0, otherwise \end{cases}$. Since each flow receives a guaranteed service independent of the interference flow, we call this Isolation Property (IP) because we treat as if the arbiter is divided into several independent parts and serves each flow separately. For 2 flows, we derive the ESC of tag flow f_1 as follows

$$\hat{\beta}_{sys\_ip} = \frac{\phi_1}{\phi_1 + \phi_2} \beta^\sigma \otimes \beta \otimes \delta_{\frac{\phi_2}{R_\sigma}} \ .$$

After derivation and simplification, the ESC turns out to be a latency-rate server with $\begin{cases} T_{sys\_ip} = T_\sigma + T + \frac{\phi_2}{R_\sigma} \\ R_{sys\_ip} = \frac{\phi_1}{\phi_1+\phi_2}(R_\sigma \wedge R) \end{cases}$ [6], where $\wedge$ is the minimum operation, $a \wedge b = min\{a, b\}$.

Then we can derive the worst-case delay bound. Since we have two arrival models for the tag flow, we give different delay bounds accordingly. For the (b, r) model, we have

$$\bar{D}_{br\_ip} = T_{sys\_ip} + \frac{b_1}{R_{sys\_ip}}, r_1 \leq R_{sys\_ip} \ .$$

For the TSPEC model, we have

$$\bar{D}_{tsp\_ip} = \frac{M + \frac{b_1-M}{p-r_1}(p - R_{sys\_ip})^+}{R_{sys\_ip}} + T_{sys\_ip}, r_1 \leq R_{sys\_ip} \ .$$

C. Analysis with Left-over Service Property

Using IP, we can find per-flow delay bound, which only depends on the allocated service and is independent of the characteristics of the interference flow. However, this may result in pessimistic result if the interference flow uses less than allocated bandwidth. In fact, with WRR, a flow can use more than its allocated bandwidth if the other flow uses less. To remedy this problem, we can compute the per-flow delay bound using the Left-over service Property (LP) [5]. The key idea with LP is that a flow's actual service depends not only on the server but also on the interference flow's characteristics.

Consider that a node serves 2 flows f_1 and f_2 with an arbitrary multiplexing, and provides service $\beta_{R,T}$ to the aggregate flow. Assume that the interference flow f_2 has arrival curve α_2. If $\hat{\beta}_{sys\_lp}(t) = [\beta_{R,T}(t) - \alpha_2(t)]^+$ is wide-sense increasing, then $\hat{\beta}_{sys\_lp}$ is a service curve for flow 1. We can prove that whether f_2 is characterized as (b, r) curve γ_{r_2,b_2}, or TSPEC curve (p, M, r_2, b_2), $\hat{\beta}_{sys\_lp}$ is always a service curve for f_1. For practical settings, we have $R \leq p = 1$, so the ESC for f_1 is $\hat{\beta}_{sys\_lp}(t) = (R - r_2)[t - \frac{b_2+RT}{R-r_2}]^+$, which means

$$\begin{cases} T_{sys\_lp} = \frac{b_2+RT}{R-r_2} \\ R_{sys\_lp} = R - r_2 \end{cases} \ .$$

And we calculate the delay bound of f_1 with the (b, r) model or the TSPEC model respectively by

$$\bar{D}_{br\_lp} = T_{sys\_lp} + \frac{b_1}{R_{sys\_lp}}, r_1 \leq R_{sys\_lp} \ ,$$

$$\bar{D}_{tsp\_lp} = \frac{M + \frac{b_1-M}{p-r_1}(p - R_{sys\_lp})^+}{R_{sys\_lp}} + T_{sys\_lp}, r_1 \leq R_{sys\_lp} \ .$$

We have now derived per-flow delay bounds using the two kinds of analysis. Taking advantage of both IP and LP, we get

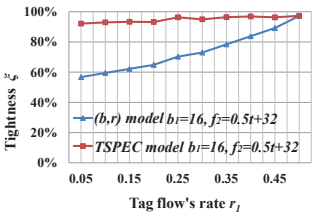

Fig. 3. Multiplexing flows: (b,r) vs. TSPEC model (by IP)

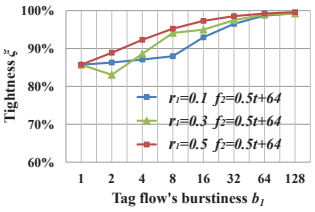

Fig. 4. Multiplexing flows: Tag flow's burstiness (by TSPEC & IP)

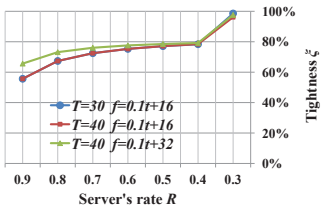

Fig. 5. Single flow: Server's rate (by TSPEC)

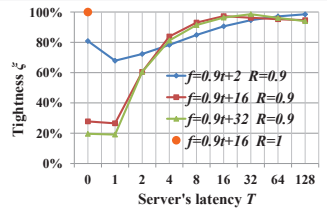

Fig. 6. Single flow: Server's latency (by TSPEC)

a combined formula $\bar{D} = h(\alpha_1, \hat{\beta}_{sys\_ip}) \wedge h(\alpha_1, \hat{\beta}_{sys\_lp})$ to get more accurate prediction of delay bound.

V. Evaluation

A. Experimental purpose and setting

1) Purpose: The purposes are multi-folded. In general, we use the simulation approach to validate if the theoretical delay bounds are tight, and to evaluate the quality of the analysis models. More importantly, we would understand when and why they are tight or not tight. To this end, we look into how the tightness will be affected by the three factors, namely, the tag flow's characteristics, the server's characteristics, and the interference flow's characteristics. Furthermore, because simulating tightness is a reachability problem, we have to understand under which system settings tight results are more likely to appear.

2) Simulation library: We build up a cycle-accurate simulation library to simulate the multiplexing behavior. *Source* builds up the arrival curve. At any time t, its output is constrained by $b + rt$. In order to generate the most bursty traffic, we use a periodic ON-OFF injection [9]. More accurately, the output can be described by the TSPEC (p, M, r, b) model with link bandwidth constraint in consideration. *Sink* is a special type of latency-rate server without output traffic. Here we implement *Arbiter* as a WRR arbiter.

3) Experimental setting: The simulated multiplexing model is shown in Fig. 2. For arrival models, we set the average injection rate $r \in (0, 1)$ packet/cycle and burstiness $b \in [1, 128]$ packet(s). As limited by the channel bandwidth in hardware, the TSPEC model has $M = 1$ packet and $p = 1$ packet/cycle in all experiments. The arbiter is work-conserving, i.e. $R_\sigma = 1$ packet/cycle and $T_\sigma = 0$ cycle, to serve the aggregated flow. For WRR scheduling, weights are allocated as $\phi_1 = \phi_2 = 1$. For the sink, when we are discussing the influence of sink's rate and latency, R and T will be assigned case by case. Otherwise, it is also supposed to be a work-conserving server $\beta_{1,0}$. For each simulation run, at least 5000 packets are transmitted for each flow.

B. Influence of tag flow's arrival curve

1) Arrival rate: In the WRR arbiter, each flow receives a minimum rate of 0.5 packet/cycle. Tag flow $f_1 = r_1 t + 16$, $r_1 \in [0.05, 0.5]$, and interference flow $f_2 = 0.5t + 32$. Here we only focus on comparing the two arrival models, so the service property is limited to IP. From results in Fig. 3, we can see that

- The TSPEC model always gives tighter result than the (b, r) model. Under the same set of settings, the larger the difference between r_1 and R_{sys_ip} (0.5 packet/cycle), the tighter TSPEC model is than the (b, r) model.
- The TSPEC model gives stable tightness close to 1. In general, the influence of r_1 to the tightness can be eliminated by introducing the TSPEC model. In contrast, ξ_{br_ip} rises along with r_1. A sweet spot occurs when the two tightness curves finally meet each other when $r_1 = R_{sys\_ip}$.

Since the TSPEC model always gives better tightness, we shall use the TSPEC model in all the following experiments.

2) Arrival burstiness: We also investigate the influence of tag flow's burstiness by three sets of experiments, which all have interference flow $f_2 = 0.5t + 64$, tag flow $f_1 = r_1 t + b_1$, where $b_1 \in [2^0, 2^7]$. In different sets, f_1 takes different amount of bandwidth allocated to it, i.e. $r_1 = 0.2R_{sys\_ip}, 0.6R_{sys\_ip}, R_{sys\_ip}$ accordingly. Theoretically, for the worst case to happen, a packet has to encounter both the longest queuing delay caused by sink and arbiter, and the processing delay caused by the packets served before it. From results in Fig. 4 we can see that

- Tightness rises along with tag flow's burstiness b_1. When there is a larger burstiness, the buffer becomes more backlogged, which gives a chance for a packet to wait for the "full" processing delay to happen.
- The curve of $r_1 = 0.5$ is generally above that of $r_1 = 0.3$, so as curve of $r_1 = 0.3$ to that of $r_1 = 0.1$. That further demonstrates when tag flow occupies more of its allocated bandwidth, the more likely a worst case to be met.

C. Influence of service curve $\beta_{R,T}$

After settling the arrival curve, we use the one-flow-one-server case to understand how the server's characteristics would impact the tightness.

1) Service rate R: Injection flow $f = 0.1t + b$, gets a service curve $\beta_{R,T}$ with $R \in [0.3, 0.9]$. When $b = 16,\ 16,\ 32$ packets, $T = 30,\ 30,\ 40$ cycles, accordingly. As shown in Fig. 5, the influence of service rate R shows actually the same trend as that of injection rate r, for which we have already discussed before. The closer r is to R, the tighter is the delay bound.

2) Processing latency T: See Fig. 6. In the experiments, we have a fixed $f = 0.9t + b$, with $b = 2,\ 16,\ 32$. Server's rate $R = 0.9$ packet/cycle and latency T changes exponentially. We discovered that a larger T may result in a tighter result, but not necessarily. For some cases, even when $T = 0$ cycle, we can get 100% tightness (for all $p = R = 1$ packet/cycle cases, see the orange point). Also, the tightest point doesn't appear at where T is largest. We conclude that, it is the ratio of T to b that affects tightness, rather than the value of T. When $T \geq b$,

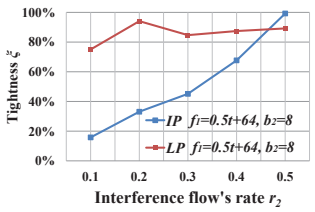

Fig. 7. IP vs. LP: impact of interference flow's rate when f_1 is busy

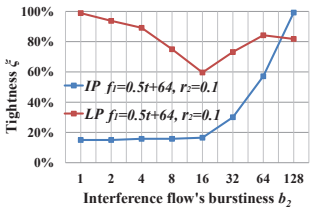

Fig. 8. IP vs. LP: impact of interference flow's burstiness when f_1 is busy

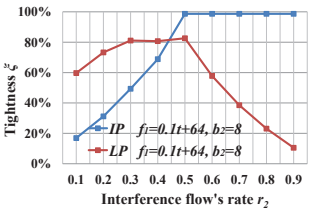

Fig. 9. IP vs. LP: impact of interference flow's rate when f_1 is less busy

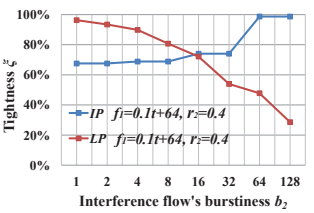

Fig. 10. IP vs. LP: impact of interference flow's burstiness when f_1 is less busy

packets have the chance to wait and accumulate before they are consumed by server, which results in a "full burstiness". When the "full burstiness" is paid by the server, the whole processing delay can be encountered.

D. Influence of interference flow's arrival curve

Now we discuss the impact of interference flow's characteristics, which relates to the actual service the tag flow can get. As discussed above, for WRR there will be more than one choice for describing the arbiter's service curve. Typically, the IP service curve and LP service curve intersect each other at a point. If no arrival curve information is provided, it cannot be determined. So, the practical method is to analyze the injected traffic patterns first, and then choose a proper service curve based on traffic situations.

1) When the tag flow is busy: In first two sets, tag flow $f_1 = 0.5t + 64$ is a comparatively busy flow, and interference flow $f_2 = r_2 t + 8, r_2 \in [0.1, 0.5]$, or $f_2 = 0.1t + b_2, b_2 \in [2^0, 2^7]$. From Fig. 7 and Fig. 8, when r_2 or b_2 gets larger, tightness of IP increases, while the one with LP does not change significantly. ξ_{tsp_lp} is dramatically higher than ξ_{tsp_ip} when r_2 or b_2 is small, but the difference lessens when r_2 or b_2 increases and finally ξ_{tsp_ip} exceeds ξ_{tsp_lp} when r_2 or b_2 is large enough. The IP model defines minimum rate for the tag flow, which is independent from the interference flow. One the other hand, the LP model gives the maximum service a tag flow can get after interference flow being extracted from the total service capacity. It takes consideration when f_1 grabs some of the service allocated to f_2. That explains why when f_2 is less busy, the leftover service model is much tighter, but when f_2 gets as busy as f_1, the IP model performs better.

2) When the tag flow is less busy: In the following sets, tag flow $f_1 = 0.1t + 64$ is a comparatively less busy flow, and interference flow $f_2 = r_2 t + 8, r_2 \in [0.1, 0.9]$, or $f_2 = 0.1t + b_2$, $b_2 \in [2^0, 2^7]$. The trend for ξ_{tsp_ip}, ξ_{tsp_lp} and their relationships hold the same as pervious experiments (See Fig. 9 and Fig. 10). An interesting point is, after a certain point, ξ_{tsp_ip} becomes saturated around 100%. In Fig. 9, it appears after $r_2 = 0.5$. That is because the interference flow has used up all its allocated service without any capacity left to f_1. In Fig. 10, it appears at $b_2 = b_1 = 64$. Because when b_2 is large enough, it can cause much blocking to f_1 in short time intervals.

VI. Conclusion

We present different arrival and service analysis models for WRR multiplexing, and show how different parameters influence delay bound tightness, giving answers to what is the quality of analysis bounds and when and why a worst case would happen. We conclude that: (1) All the analytic delay bounds are reachable. Simulated maximum delays can approach to analytic ones, giving tightness nearly 100%. (2) The TSPEC model is tighter than the (b, r) model, but much more complicated to calculate when all flows use it. Yet the (b, r) model can also deliver tight results when flow rate r is close to its equivalent service rate R_{sys}. (3) For WRR, there is no absolute winner between isolation and left-over service properties. When interference flow is less busy, using the left-over service property gives tighter bound; while when it is busy, using the isolation property is tighter. From the experiments we can see that the worst-case delay does not come "for free". They appear under specific parameter settings of flows' arrival curves and service curves.

In the future, we will extend our work to different arbitration policies, like priority based and blind arbitrations. Our aim is to draw some general conclusions for various multiplexing models.

Acknowledgment

The research is sponsored in part by Intel Corporation through a research gift. In particular, we thank Alexander Gotmanov from Intel Corporation for valuable discussions and advices.

References

[1] R. L. Cruz, "A calculus for network delay, part I: Network elements in isolation; part II: Network analysis," *IEEE Transactions on Information Theory*, vol. 37, pp. 114–131, January 1991.

[2] R. Agrawal, R. L. Cruz, C. Okino, and R. Rajan, "Performance bounds for flow control protocols," *IEEE/ACM Transactions on Networking*, vol. 7, pp. 310–323, June 1999.

[3] S. Chakraborty, S. Künzli, L. Thiele, A. Herkersdorf, and P. Sagmeister, "Performance evaluation of network processor architectures: Combining simulation with analytical estimation," *Computer Networks*, vol. 41, pp. 641–665, April 2003.

[4] C.-S. Chang, "Stability, queue length, and delay of deterministic and stochastic queueing networks," *IEEE Transactions on Automatic Control*, vol. 39, pp. 913–931, May 1994.

[5] J.-Y. Le Boudec and P. Thiran, *Network Calculus: A Theory of Deterministic Queuing Systems for the Internet*. Number 2050 in LNCS, Springer-Verlag, 2004.

[6] Y. Qian, Z. Lu, and W. Dou, "Analysis of worst-case delay bounds for best-effort communication in wormhole networks on chip," in *Proc. of The 3rd NOCS*, pp. 44–53, May 2009.

[7] IETF, "RFC 2210: The use of RSVP with IETF integrated services," September 1997. http://www.ietf.org/rfc/rfc2210.txt.

[8] D. Stiliadis and A. Varma, "Latency-rate servers: a general model for analysis of traffic scheduling algorithms," *IEEE/ACM Transactions on Networking (ToN)*, vol. 6, pp. 611–624, October 1998.

[9] Z. Lu, M. Millberg, A. Jantsch, A. Bruce, P. van der Wolf, and T. Henriksson, "Flow regulation for on-chip communication," in *Proc. of DATE 2009*, pp. 578–581, April 2009.