

Mathematical Analysis of Buffer Sizing for Network-on-Chips under Multimedia Traffic

Ahmad Khonsari
ECE Department, University of Tehran
School of Computer Science, IPM
ak@ipm.ir

Mohammad R. Aghajani, Arash Tavakkol
and Mohammad S. Talebi
School of Computer Science, IPM
{aghajani,arasht,mstalebi}@ipm.ir

Abstract—Designing appropriate buffer sizes for routers within Network-on-Chip (NoC) so as to minimize the power while preserving the required performance in the presence of self-similar traffic has been considered a challenging problem in the literature. A few analytical studies carried out in NoC modeling have been adopted assumptions such as exponentially-distributed packet inter-arrivals, and conclusions reached under such assumptions may be inappropriate in the presence of self-similar traffic. Through mathematical analysis this paper predicts the optimal buffer size under self-similar traffic using Discrete Poisson Pareto Burst Process (DPPBP). The validity of the mathematical expressions is demonstrated through simulation experiments.

I. INTRODUCTION

Recent measurements of on-chip traffic [15], [16], [21] have convincingly shown that scale-invariant burstiness (i.e. self-similarity [12]) is being recognized in both pair-wised single application [21] and over entire network [16].

As a pioneer study Varatkar and Marculescu [21] provide evidence about the presence of self-similar phenomena at the coarse-grain level in on-chip traffic generated by multimedia applications and have shown how self-similar processes can be used effectively to model the bursty traffic behavior at chip-level. The authors in [15], [16] extended the results of [21] to cycle-accurate level and proposed a stochastic traffic generator being aware that

- 1) On-chip traffic is non-stationary
- 2) On-chip traffic flows contain long range dependent behavior that must be taken into account when synthesizing traffic.

Recently, the authors in [18] propose an empirically-derived network on-chip traffic model for homogeneous NoCs. Their comprehensive model captures the spatio-temporal characteristics of NoC traffic accurately with less than 5 percent error when compared to actual NoC application traces gathered from full system simulations of three different chip platforms. namely TRIPS CMP [14], RAW CMP [19] and cache coherent CMP comprises 16 tiles each with one processor, and some memory and caches [18]. In all the above-mentioned three architectures injected network traffic possesses self-similar temporal properties [12].

Few studies target performance evaluation of multi-core and multiprocessor on-chip through mathematical analysis [5], [6], [11]. A delay model of wormhole switching has been proposed in [5]. The model assumes that routers have single flit buffers and packet size dominates the overall

latency. Recently, a mathematical performance methodology for wormhole-switched NoCs based on a novel router model has been presented in [11]. Using the router model the average number of packets at each buffer in the network as a function of the traffic arrival process has been computed. In another study [6] the buffer sizing problem in packet switched networks has been investigated and a performance model based on queuing theory has been obtained. However, this model considers only exponential packet size.

Due to the strong impacts of LRD traffic on the buffer size [21], the problem of optimal buffer sizing becomes an issue of critical importance under self-similar traffic.

This paper extends the work presented in [6] and investigates the optimal buffer size of NoCs' routers under self-similar traffic using DPPBP and through mathematical analysis we formulate the buffer overflow probability and lower and upper bound of the buffer has been investigated. The validity of the obtained expressions is demonstrated by comparing results predicted by the analytical model against those attained through simulation experiments.

The rest of the paper is organized as follows. Section II, reviews the preliminaries and main concepts related to LRD and self-similarity and then explores DPPBP and presents the analytical method for deriving the upper and lower bound of the buffer size in NoCs' routers. Through simulation study, we validate the effectiveness and accuracy of the proposed expressions in Section III. Finally, Section IV summarizes the main conclusions of the paper.

II. PERFORMANCE ANALYSIS

This section describes first the self-similar process followed by the node structure, and then the assumptions used in the analysis and presents the analytical model. Let $X = \{X_k, k = 1, 2, 3, \dots\}$ be a stationary stochastic sequence and let $X(m)$ be the corresponding aggregated sequence (with aggregation level m), obtained by averaging the original sequence X over non-overlapping blocks of size m :

$$x_k^{(m)} = \frac{1}{m} \sum_{i=(k-1)m+1}^{km} x_i; \quad k = 1, 2, 3, \dots$$

The stationary sequence X is called *exactly self-similar*, with self-similarity parameter H (the Hurst parameter [12]), if for all m , its finite dimensional distributions are identical to

those of the aggregated sequence $m^{1-H}X(m)$, that is,

$$x \doteq m^{1-H}X(m)$$

where \doteq indicates identical finite-dimensional distributions. The sequence X is called *asymptotically self-similar* if the previous condition holds as $m \rightarrow \infty$. Another less strict definition which involves the second-order moments exclusively is reported in the literature (see [17] for details). It has been shown that a second-order stationary process whose auto-correlation function decays hyperbolically is asymptotically second-order self-similar [20]. For this reason, although LRD and self-similarity are not equivalent concepts, they are often utilized without distinction and in this paper we use these terms interchangeably.

A. 2D mesh and its node structure

An $m \times n$ 2D mesh NoC consists of a set of IPs $\mathcal{V} = \{(x, y) \mid 1 \leq x \leq m, 1 \leq y \leq n\}$, where each IP (x_1, y_1) is connected to its neighbors $(x_1 \pm 1, y_1)$ and $(x_1, y_1 \pm 1)$, if they exist. The PE contains a processor and some local memory. The router has at most 5 input and 5 output channels. A node is connected to its neighboring nodes through 4 input and 4 output channels. These channels are labeled as *East*, *North*, *West* and *South* channels, respectively. The remaining channel is labeled as *Local* channel and used by the PE to inject/eject packets to/from the NoC. The router contains an address decoder, a channel controller and some flit buffers for each incoming channel that stores the input packets before delivering them to the output channels. Each such input buffer in the router can have a different depth. This can be easily implemented, for instance, at the instantiation phase through a parameterized design methodology. The FCFS input buffer is regulated through a back-pressure mechanism. Under this scheme, a packet is held in the buffer until the downstream router has enough empty space available (in the corresponding input buffer) such that network will not drop any packet in transit.

B. Assumptions

The model is based on the assumptions that have been used in [6].

- a. A 2D mesh NoC is organized in m rows and n columns with Deterministic routing [18] and Virtual-cut-through switching is also being employed. The IPs are labeled as (i, j) which corresponds to the position of the IP in a row and a column.
- b. The packet size is fixed and is equal to arbitrary number of flits so that each flit requires one-cycle transmission time across a physical channel.
- c. The buffer size is measured in multiples of packet size.
- d. The size of any input buffer must be $m \times R$, where m is a positive integer and R is the size of packets in flits.
- e. The size of the local input buffer (the input channel which accepts packets from the router's local PE) is infinite. This is a reasonable assumption since the PE can also use its local memory (which is usually much larger compared to router buffers) to store input packets.

Due to this assumption, the size of local input buffer will not be considered anymore in the allocation process. Moreover, packets are transferred to the local PE as soon as they arrive at their destinations.

Furthermore, we use the following assumption instead of exponentially-distributed packet size in [6] to capture the effects of LRD behavior.

- g. IPs generate traffic independently of each other such that the number of flows arrived at time t for each IP x, y , is distributed as a Poisson process with parameter $a_{x,y}$, and the duration of each flow follows Discrete Pareto Process with parameters c_0 (location parameter) and α (shape parameter). The obtained combined process is known as *Discrete Poisson Pareto Burst Process (DPPBP)* (details will be explained in Section II-C.1).

C. Outline of the model

Due to fixed size packet length and deterministic routing the order of packets are preserved through the network and thus, in the analysis, a packet can be treated as a basic/atomic unit since it will always be transmitted or buffered as an indivisible entity. In the absence of packet contention, the service time of each packet in a router (measured as the time span from the moment when the packet header arrives at the input channel of the router to the time it takes to receive it by the input channel of the downstream router) is fixed and can be accurately calculated. More precisely, the service time per packet (T) in a router without contention can be calculated as follows:

$$T = T_{dec} + T_r + T_{arb} + T_{xb} + T_l \quad (1)$$

In Eq. (1), T_{dec} , T_r and T_{arb} are the delays of the address decoding, routing path selection and crossbar arbitration, respectively. These parameters are usually independent of the packet size. On the other hand, T_{xb} and T_l model the delays in the crossbar and link traversal, respectively; they are usually proportional to the packet size.

1) *Traffic Modeling*: To model self-similarity, we use the model originally developed in [20] based on DPPBP. Consider IP x, y to be an IP located in position (x, y) and let $Y_{x,y}$ be the stream of packets generated by IP x, y . The packets are assigned to flows (or sessions) and thus the traffic is the aggregation of packets generated by flows. The flows are enumerated by $s \in \mathbb{Z}^+$. Throughout this paper enumeration, means that we assign a number $s \in \mathbb{Z}$ to each element of a given set S , such that the number $s = 0$ is assigned to the first element and the number $s = 1$ is assigned to the second element and so on. Each IP x, y may generate several flows s , contributing to the total traffic of NoC. Each flow s of IP x, y starts to generate its packets at time denoted by $\omega_s^{x,y}$.

By the above enumeration, the time of generating flow s is less than or equal to the time of generating flow $s + 1$ on the same IP (i.e. $\omega_s^{x,y} \leq \omega_{s+1}^{x,y}$). The moment $\omega_s^{x,y}$ is called the time at which flow s arrives. The flow s generates $R^{x,y}$ at each time $\omega_s^{x,y} + i - 1$, $i \in \{1, \dots, \tau_s^{x,y}\}$, in its "on interval" (i.e. $\omega_s^{x,y}, \dots, \omega_s^{x,y} + \tau_s^{x,y} - 1$). The number $R^{x,y}$ is called the flow rate for IP x, y and is a finite positive integer;

$R^{x,y} \in \mathbb{Z}^+$. The time interval $\omega_s^{x,y}, \dots, \omega_s^{x,y} + \tau_s^{x,y} - 1$ is called the *active (on) period* of flow s of $\text{IP}_{x,y}$ and $\tau_s \in \mathbb{Z}$ is called the length of the flow s active period. Before time $\omega_s^{x,y}$ and after time $\omega_s^{x,y} + \tau_s^{x,y} - 1$, the flow s does not generate any packets. At any time $t \in \mathbb{Z}$, more than one flow arrival can occur. Let $\xi_t^{x,y}$, denote the number of flows arrived at t , i.e. $\xi_t \in \mathbb{Z}^+$ is the number of flows started their active periods at t . Thus,

$$Y_{x,y}(t) = \sum_{s \in \mathbb{Z}^+} \theta_s^{x,y}(t - \omega_s^{x,y} + 1); \quad t \in \mathbb{Z}^+ \quad (2)$$

where $\theta_s^{x,y}(i) = R^{x,y}$ for $i \in \{1, \dots, \tau_s^{x,y}\}$ and 0 otherwise. This means that $Y_{x,y}(t)$ is the total number of packets generated by all active flows at time t . It is assumed that the random variables $\tau_s^{x,y}, s \in \mathbb{Z}^+$ are i.i.d.; the numbers of flow arrivals, $\xi_t^{x,y}, t \in \mathbb{Z}$, are i.i.d. random variables with $a_{x,y} \doteq E(\xi_t^{x,y}) < \infty$; the random variables $\tau_s^{x,y}$ are mutually independent of $\xi_t^{x,y}$ and $\omega_s^{x,y}$.

Moreover, we assume that all IPs generate the traffic independently, i.e. all random variables $\xi_t^{x,y}, \omega_s^{x,y}$ and $\tau_t^{x,y}$ are mutually independent for all $\text{IP}_{x,y}$. Furthermore, it's assumed that all IPs have the same flow rate R and have identically distributed random variables $\tau_s^{x,y}$. Let τ be a generic symbol for all $\tau_s^{x,y}$ and $\xi^{x,y}$ be a generic symbol for all $\xi_t^{x,y}, t \in \mathbb{Z}$. When τ in $Y_{x,y}$ is Pareto distributed with parameter α and $\xi_{x,y}$ is Poissonian with parameter $a_{x,y}$ we obtain the following process known as DPPBP:

$$\Pr\{\tau = l\} = c_0 l^{-\alpha-1}, \quad 0 < \alpha < 2, \quad l \in \mathbb{N} \quad (3)$$

where

$$c_0 \doteq \left(\sum_{l=1}^{\infty} l^{-\alpha-1} \right)^{-1} \quad (4)$$

and

$$\Pr\{\xi^{x,y} = k\} = e^{-a_{x,y}} \frac{a_{x,y}^k}{k!}, \quad 0 < a_{x,y} < \infty, \quad k \in \mathbb{Z}^+. \quad (5)$$

The DPPBP with parameters R, α and $a_{x,y}$ is stationary (in narrow sense) and ergodic [4], [12].

The authors in [20] has shown that a discrete-time process $Y_{x,y}(t)$ described in Eq. (2) is second-order asymptotically self-similar with parameter $H = 1 - \beta/2, 0 < \beta < 1$, if $\xi^{x,y}$ is a Poisson random variable and τ is distributed as

$$\Pr\{\tau = l\} = L(l)l^{-(\beta+2)}, \quad l \rightarrow \infty \quad (6)$$

where $L(l)$ is a slowly varying function at infinity.

If τ in Eq. (6), is a Pareto random variable, then $L(l)$ reduces to a constant which is a slow varying function at infinity and thus the conditions of second-order asymptotically self-similarity is satisfied for a DPPBP. Thus the traffic generated by $\text{IP}_{x,y}$ is a LRD process with Hurst parameter $H = 3 - \alpha/2$.

2) *Y/D/C/h Queueing System*: Consider a discrete-time system with a finite buffer and a channel that corresponds to waiting line and server in a finite buffer queueing system, respectively. In this queueing system, the time is divided into slots with the duration of one cycle per slot. Thus, a

typical slot, namely slot t spans the time interval $[t, t+1)$. Let $Y = (\dots, Y_{-1}, Y_0, Y_1, \dots)$, where Y_t is the number of packets arrived at time $t \in \mathbb{Z}$, be a renewal stochastic process representing the input arrival traffic at an input channel. The buffer has a finite size h .

The output channel can transmit (serve) no more than C packets which can be taken out of the packets waiting in the buffer and Y_t newly arrived packets. The considered queueing system is denoted as $Y/D/C/h$, where Y denotes the input traffic Y_t , D stands for the deterministic service time equal to 1 slot time, C is the number of servers and h is the buffer size. Unfortunately, the buffer size distribution for Y_t following the DPPBP has yet to be discovered. However, the boundaries of buffer overflow probability in steady state have been developed in [20].

Let random variable A_t be the buffer overflow indicator, i.e. A_t is 1 when the buffer is full at the time t and is 0 otherwise. We define

$$P_{over} = \lim_{t \rightarrow \infty} (\Pr\{A_t = 1\})$$

In [20] the lower and upper bounds of P_{over} , for the DPPBP input traffic with parameters R, α and ξ is given by:

Upper bound:

$$P_{over} \leq \bar{c} h^{(1-\alpha)k}, \quad h \rightarrow \infty, \quad k = 1 + \lfloor \frac{C}{R} - aE(\tau) \rfloor \quad (7)$$

where

$$\bar{c} = \frac{1}{k!} \left(a c_0 (\alpha - 1)^{-\alpha} \left(\frac{C}{R} + 2 \right)^{\alpha-1} R^{\alpha-1} \right)^k \quad (8)$$

Lower bound:

$$P_{over} \geq \underline{c} h^{(1-\alpha)k}, \quad h \rightarrow \infty, \quad k = 1 + \lfloor \frac{C}{R} - aE(\tau) \rfloor \quad (9)$$

where

$$\underline{c} = \frac{c_0^k R^{(\alpha-1)k}}{\alpha(\alpha-1)^k (E(\tau) + (1 - e^{\rho/E(\tau)})^{-1} - 1)} \quad (10)$$

where $a = E(\xi)$ and $c_0 = \Pr\{\tau = 1\}$. Moreover, in Eq. (10), $\rho = aE(\tau)$ if $aE(\tau) \leq 1$, otherwise ρ is any number satisfying

$$0 \leq \rho < \begin{cases} 1 + \delta - \Delta & \Delta \geq \delta \\ \delta - \Delta & \Delta > \delta \end{cases} \quad (11)$$

in which

$$\begin{aligned} \delta &= aE(\tau) - \lfloor aE(\tau) \rfloor \\ \Delta &= \frac{C}{R} - \lfloor \frac{C}{R} \rfloor \end{aligned} \quad (12)$$

We use these bounds for overflow to calculate the bounds of optimal buffer size under LRD traffic. It is noteworthy to mention that the P_{over} probability does not decay in accordance with the exponential law, which is usual in teletraffic theory, but according to the power law of h .

3) *Buffer Sizing Problem Definition*: We would like to find the buffer depth assignment for each input channel, across all of the on-chip routers, so as to minimize the average end-to-end packet delay. The inputs of the problem are the communication probability between each communicating IP pair and the total budget of buffering resources that the designer is allowed to use. Thus the problem can be formulated as:

Given: Total available buffering space B in NoC Application communication characteristics $a_{x,y}$, α and $d_{x,y}^{x',y'}$ Architecture specific packet servicing time S and routing function R .

Determine: Buffer size for each input channel $l_{x,y,dir}$ which *minimizes* the average packet latency L , which is formally expressed as below:

$$\min L \quad (13)$$

subject to:

$$\sum_x \sum_y \sum_{dir} l_{x,y,dir} \leq B \quad (14)$$

over:

$$l_{x,y,dir}.$$

We use the overflow probability boundaries presented in the above section to predict the performance bottleneck channel in the following section.

4) *Solving the Buffer Sizing Problem*: As mentioned in [6] the buffer allocation algorithm starts with the minimum buffer size configuration (i.e. one packet) and iteratively increases the buffer size of the bottleneck channels until the specified value of the buffer budget is reached. The main challenging problem which makes the problem more complicated than [6] is devising a technique to detect the performance bottleneck among the different router channels, which highly affects the behavior of overflow probability in terms of the router buffer size due to high variability of self-similar traffic.

5) *Router/channel analytical models*: Given the current buffer size configuration, the algorithm tries to identify the channels where adding extra buffering space leads to the maximum improvement in performance. A set of nonlinear equations derived from $Y/D/C/H$ queuing model presented in the previous section is used to analyze the current buffer size configuration and then to detect the performance bottlenecks in the router channels. The basic idea is that, given the system configuration (which includes the traffic pattern and the size of each FIFO in the current solution), the algorithm detects the FIFO which has the highest probability to be in the *full state*. The channel which owns this particular FIFO becomes the real performance bottleneck in the current configuration and thus its size should be increased. Now, we calculate P_{over} for all input channel $C_{x,y,dir}$ of all routers $R_{x,y}$ and find the bottleneck. Each router channel $C_{x,y,dir}$ is modeled as a finite queue of length $l_{x,y,dir}$, with the input traffic $Y_{x,y,dir}$, and the service rate $\mu_{x,y,dir}$. To derive the overflow probability, the traffic arrival process and the mean service time at a channel have to be determined first. The

mean traffic rate is calculated as follows. The total arrival, $Y_{x,y,dir}$, at channel $C_{x,y,dir}$ is the sum of all flows in NoC traversing this channel and is given by:

$$Y_{x,y,dir} = \sum_{i,j} \sum_{i',j'} Y_{i,j}^{i',j'} R(i,j,i',j',x,y,dir) \quad (15)$$

In the above equation, $Y_{i,j}^{i',j'}$ denotes the fraction of the traffic of source (i,j) to destination (i',j') . The routing function $R(i,j,i',j',x,y,dir)$ determines the traffic traversing from source (i,j) to destination (i',j') , which is 1 if the routing path passes through $C_{x,y,dir}$ and is 0 otherwise. The following two theorems known as splitting and superposition are constructive in calculating $Y_{x,y,dir}$.

Theorem 1: Let Y be the stream of packets generated by source s . The destinations of packets are distributed on other nodes such that the packets pass through link l_1 with probability p and through other links with probability $q = 1 - p$. Let Y_1 be the fraction of Y passing through l_1 . If Y is a DPPBP process with parameters R , α and a , Then Y_1 is also a DPPBP with parameters R , α and pa .

Proof: The proof is omitted due to space limit. We refer the interested reader to [7]. ■

Theorem 2: Let $Y_i, i = 1, 2, \dots$ be a DPPBP process with parameters R, α and a_i and Y be the superposition of all Y_i : $Y = \sum_{i=1}^N Y_i$. Then Y is a DPPBP process with parameters R, α and $a = \sum_{i=1}^N a_i$.

Proof: The proof is omitted due to space limit. We refer the interested reader to [7]. ■

For fixed i, j, i', j' , since $Y_{i,j}$ is DPPBP, it follows from Theorem 1 that $Y_{i,j}^{i',j'}$ is also a DPPBP with parameters R, α and $a_{i,j} d_{i,j}^{i',j'}$. Moreover, Eq. (15) can be rewritten as:

$$Y_{x,y,dir} = \sum_{(i,j,i',j') \in SD_{x,y,dir}} Y_{i,j}^{i',j'} \quad (16)$$

where $SD_{x,y,dir}$ is the set of all source-destination pairs in NoC, whose traffic crossing the channel $C_{x,y,dir}$. In other words, $SD_{x,y,dir} = \{(i,j,i',j') \mid R(i,j,i',j') = 1\}$. Again, by virtue of Theorem 2 when all $Y_{i,j}^{i',j'}$ follow DPPBP, their superposition $Y_{x,y,dir}$ is also a DPPBP process with parameters R, α and $a_{x,y,dir}$, where

$$\begin{aligned} a_{x,y,dir} &= \sum_{(i,j,i',j') \in SD_{x,y,dir}} a_{i,j} d_{i,j}^{i',j'} \\ &= \sum_{i,j} \sum_{i',j'} a_{i,j} d_{i,j}^{i',j'} R(i,j,i',j',x,y,dir) \end{aligned} \quad (17)$$

Considering $Y/D/C/h$ queuing model described in the previous section and using Eq. (7) and Eq. (9), buffer overflow probability $P_{x,y,dir}$ can be written as

$$\underline{c}_{x,y,dir} l_{x,y,dir}^{(1-\alpha)k_{x,y,dir}} \leq P_{x,y,dir} \leq \bar{c}_{x,y,dir} l_{x,y,dir}^{(1-\alpha)k_{x,y,dir}} \quad (18)$$

where

$$k_{x,y,dir} = 1 + \left\lfloor \frac{\mu_{x,y,dir}}{R} - a_{x,y,dir}E(\tau) \right\rfloor \quad (19)$$

and $\mu_{x,y,dir}$ is the service rate for channel $C_{x,y,dir}$ in the presence of contention, $\underline{c}_{x,y,dir}$ and $\bar{c}_{x,y,dir}$ are the constants defined in Eq. (10) and Eq. (8) upon substituting a, C, h and k by $a_{x,y,dir}, \mu_{x,y,dir}, l_{x,y,dir}$ and $k_{x,y,dir}$, respectively.

Now let us calculate $\mu_{x,y,dir}$ which is not trivial, as it depends not only on the router's service delay, but also on probabilities of a packet being routed to each downstream channel and whether or not the downstream channels are full. For instance, if the packet is to be delivered eastward and $C_{x+1,y,W}$ is full, then the packet has to wait in $C_{x,y,N}$. Using the method presented in [6], we can write the following expression for service rate $\mu_{x,y,dir}$ at a channel:

$$\begin{aligned} \mu_{x,y,dir} &= RE(\tau)a_{x,y,dir} \\ &+ \frac{1}{\frac{1}{T-RE(\tau)a_{x,y,dir}} + \frac{1}{\bar{\mu}_{x,y,dir}-RE(\tau)a_{x,y,dir}}} \end{aligned}$$

where T is given by Eq. (1) and

$$\bar{\mu}_{x,y,dir} = \sum_{dir'} \bar{\mu}_{x,y,dir}^{dir'} P_{x,y,dir}^{dir'} \quad (20)$$

and $P_{x,y,dir}^{dir'}$ is the probability that an incoming packet from channel $C_{x,y,dir}$ leaves the router using direction dir' . As discussed in [6], this parameter is predetermined. Moreover, $\bar{\mu}_{x,y,dir}^{dir'}$ is the service rate due to contention corresponding to the traffic originated by $C_{x,y,dir}$ and passing through the router (x, y) to the outgoing direction dir' and is a function of the transmission rate of input and output channels and buffer overflow probability of the neighboring router. For example service rate for East channel can be written as

$$\bar{\mu}_{x,y,N}^E = \frac{1}{P_{x+1,y,W}} - a_{x+1,y,W} + P_{x,y,N}^E a_{x,y,N} \quad (21)$$

The service rate of the other directions follows the same rule. The set of the above non-linear equations for all routers can be solved iteratively to obtain the buffer overflow probability for all nodes and all directions which finally determine the bottleneck link.

III. EXPERIMENTAL RESULTS

Numerous validation experiments have been performed for several combinations of network size, packet size, and different self-similar input traffic parameters. However, for the sake of specific illustration, the following results are only presented. Fig. 1 presents the buffer overflow probability against buffer size using the Markovian and upper and lower bound self-similar queueing system denoted by $M/M/1/h$, $Y/D/C/h$ upper bound and $Y/D/C/h$ lower bound, respectively. The figure shows that the analytical predicted boundaries enclose the simulation data very well. The obtained accuracy under the proposed method is below %20 which is much better than %35 accuracy.

Fig. 2 depicts buffer overflow probabilities results provided by simulator and the upper bound predicted by analytical model against the buffer size. Two Hurst parameters,

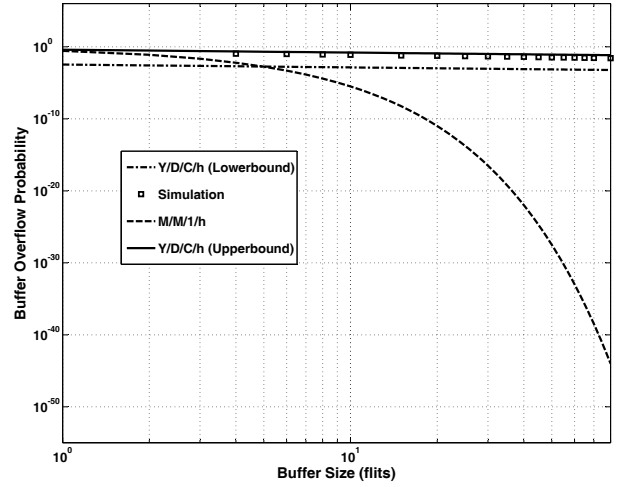


Fig. 1. Log-Log Plot of Buffer Overflow Probability vs. Buffer Size for Markovian and Upper Bound and Lower Bound of Bursty Queueing Systems and Simulation Results

namely $H = 0.6$ and $H = 0.8$ which is reported in measurements study [11] and are typical in other studies, are considered over multiple time scales.

Having validated the analytical model, let us assess the impact of self-similarity on NoC. Figures divulge that self-similarity has a significant effect on router buffer size especially in NoCs with scarce resources and renders the developed models of Poissonian traffic of less value for MP-SoCs targeting multimedia applications. Fig. 3 demonstrates that increasing channel service rate reduces self-similarity degree and thus has a higher impact than increasing the router buffer size. The lower bound has not been illustrated in Fig. 2 and Fig. 3, since it has very small value and the simulation results were higher than this threshold for all the implemented experiments.

Counterintuitive, our experiments show that in small meshes the IPs at the borders are bottleneck and degrade the performance heavily. This is due to the fact that the effective number of channels of these IPs' are less than the IPs within the network. We have found that the traffic generated by the border IPs in small mesh, i.e. 3×3 , cause the output channels become busy and thus as an alternative viable solution we suggest to increase the capacity of the output channels of the border IPs.

IV. CONCLUSION

In this paper we have extended the work in [6] to evaluate the effects of self-similar traffic on buffer size of MPSoCs routers using DPPBP. Through comparisons between analytical bounds and extensive simulation results, we have validated the effectiveness and accuracy of the presented expressions and the upper and lower bound of buffer size. We have shown in small meshes with Deterministic routing the nodes positioned on the borders are bottleneck in terms of buffer size due to overloading by their own LRD traffic. As a supplementary result we demonstrated the effect of

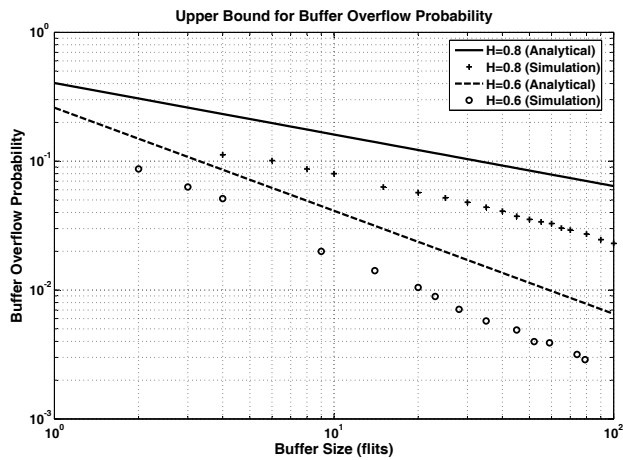


Fig. 2. Log-Log Plot of Comparison of Analytical and Simulation Results for Upper Bound of The Buffer Overflow Probability vs. Buffer Size

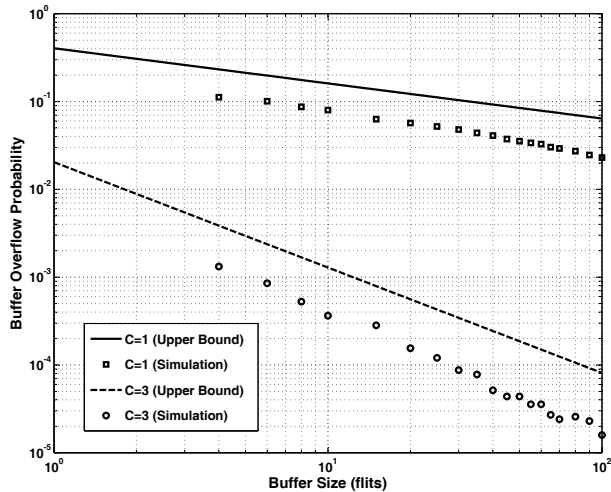


Fig. 3. Log-Log Plot of Effect of Service Rate C on The Buffer Overflow Probability vs. Buffer Size (Self-Similarity Degree $H = 0.8$)

increasing channel service rate in reducing the self-similarity degree that renders to a more predictable buffer behavior which makes the buffer allocation algorithm less complicated. This is an important result in NoC design, since the alternating solution which is increasing the router buffer size is inappropriate and greatly increases the packets delay in the NoC. Due to the high variability of LRD traffic, we require to devise innovative mechanisms such as congestion control to alleviate the degrading effects of self-similar traffic.

REFERENCES

- [1] L. Benini, G. DeMicheli, "Networks on Chips: A New SoC Paradigm", *IEEE Computer*, 35 (1), pp. 70-78, 2002.
- [2] M. E. Crovella, A. Bestavros, "Self-similarity in World Wide Web traffic: Evidence and possible causes", *IEEE/ACM Trans. Netw.*, 5 (6), pp. 835-846, 1997.
- [3] W. J. Dally, B. Towles, "Route Packets, Not Wires: On-Chip Interconnection Networks", *Design Automation Conference*, pp. 684-689, 2001.
- [4] P. Doukhan, G. Openheim, and M. Taqqu. *Theory and Applications of Long Range Dependence*. Birkhuser Boston, 2002.

- [5] Z. Guz, E. Bolotin, I. Cidon, R. Ginosar, A. Kolodny, "Efficient link capacity and QoS design for wormhole network-on-chip", *Design Automation Conference*, 2006.
- [6] J. Hu, U. Y. Ogras, R. Marculescu, "System-level buffer allocation for application-specific networks-on-chip router design," *IEEE Trans. on Computer-Aided Design of Integrated Circuits and Systems*, 25(12), 2006.
- [7] A. Khonsari, R. Aghajani, A. Tavakkol and M. S. Talebi, "Mathematical Analysis of Buffer Sizing for Network-on-Chips under Multimedia Traffic", *Technical Report No. TRCS-87121-01*, 2008.
- [8] W. E. Leland, M. S. Taqqu, W. Willinger, D. V. Wilson, "On the self-similar nature of Ethernet traffic (extended version)", *IEEE/ACM Trans. Netw.*, 2 (1), pp.1-15, 1994.
- [9] J. Lopez-Ardao, C. Lopez-Garcia, A. Suarez-Gonzalez, M. Fernandez-Veiga, R. Rodriguez-Rubio, "On the Use of Self-Similar Processes in Network Simulation", *ACM Trans. on Modeling and Computer Simulation (TOMACS)*, 10(2), pp.125-151, 2000.
- [10] I. Norros, "A storage model with self-similar input", *Queueing Systems*, pp. 387-396, 1994.
- [11] U. Y. Ogras, R. Marculescu, "Analytical Router Modeling for Networks-on-Chip Performance Analysis", *Design, Automation and Test in Europe*, 2007.
- [12] K. Park, W. Willinger, *Self-similar network traffic and performance evaluation*. John Wiley and Sons, 2000.
- [13] V. Paxson, S. Floyd, "Wide-area traffic: The failure of Poisson modeling", *IEEE/ACM Trans. Netw.*, 3 (3), pp. 226-244, 1995.
- [14] K. Sankaralingam et al., "Exploiting ILP, TLP, and DLP with the polymorphous TRIPS architecture", *30th International Symposium on Computer Architecture (ISCA-30)*, pp. 422-433, 2003.
- [15] A. Scherrer, A. Fraboulet, T. Risset, "Analysis and Synthesis of Cycle-Accurate On-Chip Traffic with Long-Range-Dependence", *Technical report 2005-53*, LIP, ENS-Lyon, 2005.
- [16] A. Scherrer, A. Fraboulet, T. Risset, "Long-Range Dependence and On-chip Processor Traffic", *ReCoSoc: Reconfigurable Communication-centric SoCs*, 2007.
- [17] O. Sheluhin, S. Smolskiy, A. Osin, *Self-Similar Processes in Telecommunications*. John Wiley and Sons, 2007.
- [18] V. Soteriou, W. Hangsheng, L. Peh, "A Statistical Traffic Model for On-Chip Interconnection Networks", *MASCOTS 2006*, pp. 104 - 116, 2006.
- [19] M. B. Taylor et al., "Evaluation of the Raw microprocessor: an exposed wire-delay architecture for ILP and streams", *31st International Symposium on Computer Architecture (ISCA-31)*, pp. 2-13, 2004.
- [20] B. Tsybakov and N. D. Georganas, "Overflow and losses in a network queue with a self-similar input", *Queueing Systems*, 35 (1-4), pp. 201-235, 2000.
- [21] G. Varatkar, R. Marculescu "Traffic Analysis for On-chip Networks Design of Multimedia Applications", *IEEE Trans. on Very Large Scale Integration (VLSI) Systems*, 12 (1), pp. 108-119, 2004.