

# Accurate Analytical Modeling for NoCs with Hybrid Arbitration under High Traffic Injection

Rahul Tripathy<sup>1</sup>, Mohammad Majharul Islam<sup>2</sup>, Riad Akram<sup>2</sup>, Raid Ayoub<sup>2</sup>, Sumit K. Mandal<sup>1</sup>

<sup>1</sup>Department of Computer Science and Automation, Indian Institute of Science, Bangalore, India

<sup>2</sup>Intel Corporation, Hillsboro, OR

**Abstract**—Analytical performance modeling of Networks-on-Chip (NoC) are important for fast design space exploration and quick pre-silicon evaluation. Existing NoC performance analysis techniques assume certain micro-architectural details (e.g., a particular arbitration technique) to be homogeneous across the entire NoC. However, emerging NoC architectures may have hybrid arbitration across the NoC to ensure high throughput. Moreover, existing analytical models estimating performance of NoCs with finite buffers fail to analyze the performance of the NoC accurately under high traffic injection which occur in several modern-day server as well as client applications. In this work, we propose a performance analysis technique for NoCs with hybrid arbitration under high traffic injection. We propose a novel transformation to accurately compute the waiting time of the queues under hybrid arbitration. We also develop a technique to compute the effective arrival statistics to the queues when the desired injection rate is high. Thorough experimental evaluation with a wide range of injection rates at the queues of an industrial NoC show that our proposed analytical model incurs only 7% error on average and 3 orders of speed-up with respect to cycle-accurate simulation under high traffic injection.

## I. INTRODUCTION

With increasing demand for high-performance computers and growing complexity of applications, there is a growing need for systems-on-chip (SoCs) with new architectures delivering higher performance. State-of-the-art SoCs for high performance computers consist of multiple networks-on-chip (NoCs) with different micro-architectures which connect multiple processing cores and other IP blocks of the SoC. While a new SoC is being developed, it needs to go through rigorous performance evaluations. The performance evaluation process usually includes cycle-accurate simulators. Prior work reveals that up to 70% of the full-system simulation time is consumed by cycle-accurate simulation of NoCs [1]. Therefore, several prior works [2–4] have proposed fast and accurate NoC performance analysis techniques to determine average waiting time of the packets in the NoC which can potentially augment the existing cycle-accurate NoC simulators [5–7].

There exist several recently emerged applications which show high compute and memory-intensiveness [8]. With new applications being executed, existing NoC arbitration techniques may not be providing optimal throughput. Thus, recently emerged SoCs consist of NoCs with hybrid arbitration [9, 10]. In hybrid arbitration, depending on whether a class of packets have strict priority or the same priority with respect to another class (depending on the destination agent), the arbiter may follow priority as well as round-robin arbitration. No prior NoC analytical modeling techniques consider hybrid arbitration. Moreover, due to the intensive nature of the

applications, compute cores as well as the memory elements in the SoC often communicate at a much higher rate. Therefore, the injection rates at the queues of the NoC connecting those processing elements are usually high. High injection rate to the NoC queues result in congestion over the entire NoC. Even if a throttling mechanism is in place, we need to analyze the effective performance of the NoC under high traffic injection. Existing analytical models for finite buffers are not accurate in the region of high injection rates.

In this work, we propose an analytical modeling approach for NoCs with arbiters having hybrid arbitration which can also operate under high traffic injection. We first propose a novel transformation which transforms a given arbiter with hybrid arbitration into an equivalent queuing system. We then present an analytical model for the transformed queuing system. The average latency of the packets in the modified queuing system is approximated as the average latency of the packets in the original queuing system. Then we compute the probability of the queues being full. We use the probability of the queues being full to construct the analytical model of the average waiting time of the packets in the NoC. To handle high traffic injection, we propose a novel technique to compute the statistics (first and second order moment) of the effective injection rate at the NoC queues. Thorough experimental evaluation with synthetic traffic as well as real applications executing on an industrial NoC show that our proposed analytical model incurs only 7% error on average with respect to cycle-accurate simulators. Following are the major contributions of our work.

- Analytical performance model for NoCs with hybrid arbitration,
- Accurate analysis of waiting time of packets in the queue under high traffic injection,
- Rigorous experimental evaluation showing the accuracy of our proposed analytical model.

## II. RELATED WORK

Round-robin arbitration provides fairness to all flows, despite the difference in types of flits. Priority arbitration provides lower latency to flits of a certain flow, but there can only be strict priority between two flows. To address this, recent NoCs [9, 10] consist of hybrid arbitration where there is no strict priority between two or more flows. Here, more than one flow can have the same priority, with round-robin arbitration between flows of the same priority. Hybrid arbitration proves to be an important innovation that addresses the bandwidth issue present in both round-robin and priority based arbitration.

TABLE I  
COMPARISON OF PRIOR RESEARCH AND OUR NOVEL CONTRIBUTION

Research	Arbitration	Finite Queue	High Traffic Injection
Xiaolong et al. [15]	Priority	No	No
Narayana et al. [1]	Priority	Yes	No
Ogras et al. [2]	Round-Robin	No	No
Mandal et al. [4]	Round-Robin	Yes	No
<b>This Work</b>	<b>Hybrid</b>	<b>Yes</b>	<b>Yes</b>

However, to the best of our knowledge, no method exists for modeling the performance of NoCs with hybrid arbitration.

Prior work proposed analytical performance models that discuss NoCs with round-robin arbitration [2, 11–13]. Authors in [2] calculate the waiting time using a contention matrix for the different flows. They assume a geometric distribution for the arriving packets in the NoC. This assumption is invalid when there are tandem queues, as the departure from a server may not follow geometric distribution. Fisher et. al computes the mean service time in an effort to calculate the total waiting time [11]. However, they fail to consider the second order moment of the service time, and thus is inapplicable here. Authors in [12] provide a waiting time model for round-robin arbitration, but assume an infinite buffer size. A support vector regression-based analytical model is proposed in [13]. However, all these techniques fail to model the NoC at high traffic injection.

Prior work proposed analytical performance models for NoCs with priority arbitration [3, 14, 15]. Authors in [14] propose a discrete-time priority-aware performance analysis technique. This method has high complexity and thus not practical for large systems. Authors of [15] consider multiple traffic classes in an NoC with priority arbitration, using two assumptions that high priority packets preempt the lower priority packets, and the Empty Buffer Approximation (EBA). Both these assumptions are not applicable for real time systems as any lower priority flit will cause extra residual time on all flits that have a higher priority than it, and the effect of the buffer filling up must be considered. Authors in [3] model NoCs with multiple priority traffic classes. However, all the previous methods fail to model the NoC under high traffic injection.

This paper presents the first performance analysis technique for NoCs with hybrid arbitration, taking into account finite buffer size, with special consideration for high traffic injection. The proposed technique first converts hybrid priority queues into tandem queuing networks, after which queue decomposition is used to find the waiting time in each queue. This approach is faster than cycle-accurate simulators, accurate, and is scalable. The comparison between existing techniques and this work are summarized in Table I.

### III. BACKGROUND AND OVERVIEW

#### A. Background on Systems with Multiple Queues

Figure 1(a) and Figure 1(b) show systems of queues with priority arbitration and round-robin arbitration with finite

TABLE II  
LIST OF THE IMPORTANT PARAMETERS USED IN THIS WORK

$J$	Number of queues
$Q_j, Q_s$	Queue- $j$ , Queue-sink
$\lambda_j, \hat{\lambda}_j$	Injection rate and modified injection rate to $Q_j$
$C_{aj}$	Squared coeff. of variation of inter-arrival time of $Q_j$
$\hat{C}_{aj}$	Modified $C_{aj}$
$T, \hat{t}_j$	Mean service time and modified mean service time of $Q_j$
$C_s$	Squared coeff. of variation of service time
$N$	Buffer size
$\rho_j$	Mean ( $\rho_j = \lambda_j T$ ) server utilization of $Q_j$
$\hat{\rho}_j$	Modified $\rho_j$
$p_{fj}$	Probability of $Q_j$ being full
$n_j$	Mean occupancy of $Q_j$
$\hat{R}_j$	Residual time of $Q_j$
$W_j$	Mean waiting time for packets in $Q_j$

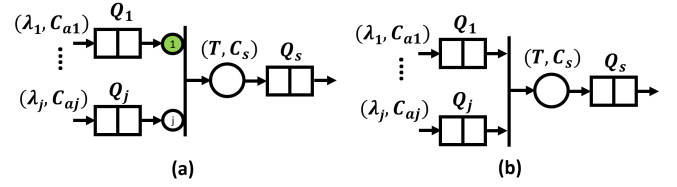


Fig. 1. Queuing System with (a) Priority arbitration and (b) Round-Robin arbitration.

buffers respectively. There are  $J$  queues which are arbitrated and each of them holds packets of different classes. We denote packets of a unique source and destination pair in an NoC as a class. The queues are denoted by  $Q_j$ , where  $1 \leq j \leq J$ . The arrival process of class- $j$  is denoted by  $(\lambda_j, C_{aj})$ , where  $\lambda_j$  is the packet injection rate and  $C_{aj}$  is the squared coefficient of variation of inter-arrival times of class- $j$ , as described in Table II. The server properties are denoted by  $(T, C_s)$  where  $T$  is the average service time and  $C_s$  is the squared coefficient of variation of service time. The output of the arbiters are connected to a downstream queue ( $Q_s$ ) which functions as a sink, where packets will wait after exiting the server. The capacity of the queues is  $N$ , and after it fills up no packet will be able to enter, stalling them and leading to back-pressure to the upstream queues. Therefore, the effective injection rate ( $\hat{\lambda}_j$ ) is different than the desired injection rate ( $\lambda_j$ ). The utilization of the server due to class- $j$  is given by  $\rho_j = \hat{\lambda}_j T$ . The system reaches saturation when the total utilization ( $\rho$ ) of the server is 1, where  $\rho = \sum_{j=1}^J \rho_j$ . Typically, at high traffic injection,  $\rho$  tends to become greater than 1. *Since at steady state  $\rho$  can never be greater than 1 due to throttling mechanism, the effective injection rate ( $\hat{\lambda}_j$ ) of some or all classes become less than the original injection rate, i.e.,  $\hat{\lambda}_j \leq \lambda_j$ .* When the queues are full, probability that the queues are full ( $p_f$ ) is used to calculate the effective injection rate. The technique to obtain the waiting time is iterative as proposed in [4]. However, the existing technique fails to estimate the waiting time when the total utilization of the server  $\rho$  is mathematically greater than 1. Moreover, they are proposed in the scenario when there is only one type of arbitration in the whole NoC. Our objective is to compute the average waiting time ( $W_j$ ) of the packets in  $Q_j$  served by an

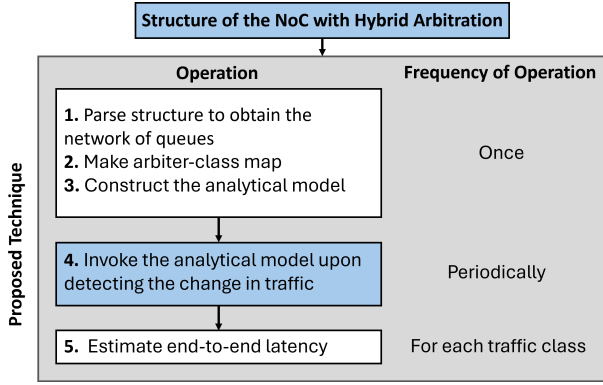


Fig. 2. Overview of the proposed technique.

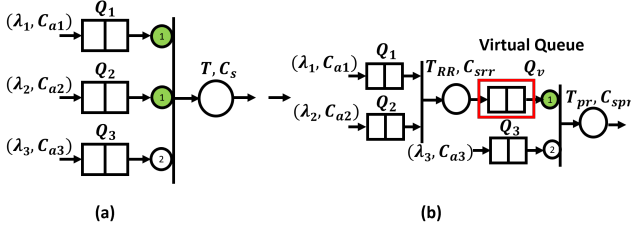


Fig. 3. (a) Hybrid priority queue and (b) its equivalent tandem queue.

arbiter at high traffic injection.

### B. Overview of the Proposed Technique

The primary goal of our work is to reduce the time taken by full-system simulations [6] by replacing cycle-accurate NoC simulations with accurate and fast analytical models. Figure 2 shows the overview of our proposed framework. The structure of the NoC with hybrid arbitration is provided to the framework. The NoC structure is first parsed to obtain the network of queues and statistics of traffic at each queue. Then a map between each arbiter and traffic class is constructed. Next, the analytical model is constructed by using the map. The analytical model is periodically invoked. Specifically, whenever any change in traffic from the host is detected, the waiting time of each traffic class is computed by executing the proposed analytical model. Therefore, our model replaces the cycle accurate simulation of NoCs.

## IV. PROPOSED ANALYTICAL MODEL

In this work, we propose an analytical model for a queuing system with hybrid arbitration with consideration of high traffic injection. First, we describe the technique to compute the average waiting time of the packets in a queuing system with hybrid arbitration. At high traffic injection, the effective injection rates at the queues are less than the desired (original) injection rate. Therefore, we then describe our proposed technique to compute the effective injection rate and compute the average waiting time of the packets in the queue.

### A. Modeling for Queuing System with Hybrid Arbitration

Figure 3(a) shows a queuing system with hybrid arbitration. In this system, packets in  $Q_1$  (class-1) and in  $Q_2$  (class-2) are

arbitrated with round-robin arbitration. Packets in  $Q_3$  (class-3) have lower priority than both class-1 and class-2. NoCs with hybrid arbitration are prevalent in emerging industrial SoCs [9]. No prior work proposed performance analysis techniques for a system of queues with hybrid arbitration.

We propose a transformation to obtain the analytical model of the waiting time of classes in a queuing system with hybrid arbitration. Figure 3(b) shows our proposed transformation. In the transformed network, we first have a round-robin arbitration which arbitrates class-1 and class-2 packets. The packets which win the arbitration are put in a virtual queue ( $Q_v$ ; shown in red rectangular box). The packets in the virtual queue  $Q_v$  and the physical queue  $Q_3$  are arbitrated through priority arbitration. The first and second order moments of the service time of the priority and the round-robin arbitration is set to the first and second order moment of the service time of the original server (in Figure 3(a)). Since the packets arrive in  $Q_v$  after getting arbitrated by the round-robin arbitration, the arrival statistics ( $\lambda_v, C_{av}$ ) to  $Q_v$  is not same as the original packet arrival statistics. We compute the arrival statistics to  $Q_v$  through the decomposition technique [16].

$$\lambda_v = \lambda_1 + \lambda_2, \quad \rho_{RR} = \lambda_v \times T_{RR}, \quad C_a = \frac{\lambda_1 C_{a1} + \lambda_2 C_{a2}}{\lambda_1 + \lambda_2}$$

$$C_{av} = \rho_{RR}^2 (C_s + 1) + (1 - \rho_{RR}) C_a + \rho_{RR} (1 - 2\rho_{RR}) \quad (1)$$

Next, we use  $\lambda_v$  and  $C_{av}$  to obtain the average waiting time for the packets at  $Q_v$  ( $W_{Q_v}$ ) and  $Q_3$  ( $W_3$ ). We utilize the technique described in Section IV-C to compute the waiting time of the packets in all the queues at high traffic injection. Finally, the waiting time of class-1 ( $W_1$ ) and class-2 ( $W_2$ ) are computed as  $W_1 = W_{Q_1} + W_{Q_v}$  and  $W_2 = W_{Q_2} + W_{Q_v}$  respectively, where  $W_{Q_1}$  and  $W_{Q_2}$  are the waiting time of the packets in  $Q_1$  and  $Q_2$  respectively.

### B. Computing Effective Injection Rate ( $\hat{\lambda}_j$ ) at High Traffic

**Priority Arbitration:** Lets consider the queuing system with priority arbitration shown in Figure 1(a). In this system, there are  $J$  queues and each of the queues consist of a particular class of packets. Packets of class- $j$  have higher priority than the packets of class- $k$  if  $j < k$ . At high injection rate, the injection rate of the highest class ( $\lambda_1$ ) is unaffected if  $\rho_1$  is less than 1, i.e.,  $\hat{\lambda}_1 = \lambda_1$ . If  $\rho_1 \geq 1$ , then  $\hat{\lambda}_1$  takes a value at which  $\hat{\rho}_1 = 1$ . The effective injection rate of the lower priority classes ( $\hat{\lambda}_j, 1 < j \leq J$ ) is computed as the maximum value it can take given the total effective utilization till that class is less than or equal to 1, i.e.,  $\sum_{i=1}^j \hat{\rho}_i \leq 1$ . Therefore,  $\hat{\lambda}_j$  is expressed as

$$\hat{\lambda}_j = \begin{cases} \frac{1}{T} \min(1, \rho_j), & j = 1 \\ \min(\lambda_j, (1 - \frac{1}{T} \sum_{i=1}^{j-1} \hat{\rho}_i)), & 1 < j \leq J \end{cases} \quad (2)$$

**Round-Robin Arbitration:** Lets consider the queuing system with round-robin arbitration shown in Figure 1(b). In this system, there are  $J$  queues which are arbitrated in round-robin fashion and each of the queues consist of a particular class of packets. At high traffic injection, the server utilization of each class tries to equalize, as the service prefers all classes

fairly. The total utilization  $\rho$  in this case is mathematically 1. Thus, class- $j$  with  $\lambda_j < 1/J$ , where  $J$  is the total number of classes, will have their injection rate unaffected while the rest of the classes split the remaining utilization equally between themselves. Therefore, the effective injection rate of class- $j$  ( $\hat{\lambda}_j$ ) is computed as

$$\hat{\lambda}_j = \begin{cases} \lambda_j, & \lambda_j \leq 1/J \\ \left( \frac{1 - \sum_{k \in K} \lambda_k}{J - K} \right), & \lambda_j > 1/J \end{cases} \quad (3)$$

where  $K$  is the set of classes with  $\lambda_k < 1/J$ . We assume that  $\lambda_j$  and  $\hat{\lambda}_j$  have the same distribution. In both cases,  $C_a$  is computed by considering bursty traffic, according to [17].

### C. Computing Average Waiting Time ( $W_j$ ) for Hybrid Arbitration under High Traffic Injection

This section describes our proposed technique to compute the average queue waiting time using the modified injection rate for hybrid arbitration under high traffic injection. First, the queuing system with hybrid arbitration is modified following the technique proposed in Section IV-A. The occupancy of class- $j$  in a queue with given statistics of arrival and service is computed by the following equation [18].

$$n_j = \frac{\hat{\rho}_j(\hat{\rho}_j - 1 + \hat{C}_{aj} + \hat{\rho}_j\hat{C}_{aj} + \hat{\rho}_j\hat{C}_{sj})/2}{1 - \hat{\rho}_j} + \hat{\rho}_j \quad (4)$$

In this equation,  $\hat{\rho}_j = \hat{\lambda}_j \hat{t}_j$ .  $\hat{t}_j$  is the modified service time of class- $j$  packets. The modified service time for priority arbitration and round-robin arbitration is obtained through the technique described in [1] and [4] respectively. Equation 4 can also be expressed as

$$n_j = \frac{\hat{\lambda}_j R_j}{(1 - \hat{\rho}_j)} + \hat{\rho}_j \quad (5)$$

where  $R_j$  is the residual time of class- $j$ . Residual time is defined as the remaining service time of the packet under a service as seen by a new arrival into the queue [19]. When a queuing system is operating under high traffic injection, the queues will likely be full almost all the time since all the queues are finite in size. Whenever the server is finished servicing a packet, the packet at the head of the queue is admitted into the server and a new packet is admitted into the end of the queue from the upstream system. Therefore, the new packet will experience a residual time which is always equal to the average service time of the server ( $T$ ). Hence, under high traffic injection,  $R_j = T$ . This means that every packet in the buffer will advance forward every  $R_j$  cycles. Equating Equation 4 and Equation 5 to get  $R_j$  and substituting  $\hat{\rho}_j$  as 1 in  $R_j$ , we obtain

$$W_j = \left( \frac{N(2\hat{C}_{aj} + \hat{C}_{sj}) + 2}{2\hat{\lambda}_j} \right) \quad (6)$$

as the mean waiting time for  $Q_j$  for priority arbitration. For round-robin arbitration, we add a correction term to  $W_j$  required due to the assumption on the residual time.

$$W_j = (a_0N + a_1) \left( \frac{N + (2 - N)\hat{C}_{aj}}{2N\hat{\lambda}_j} \right) \quad (7)$$

---

### Algorithm 1: End-to-end Latency Estimation

---

```

1 Input: Queuing network, service process ( $T, C_s$ ),
   arrival statistics for each class ( $\lambda, C_a$ ), buffer size
2 Output: Average end-to-end interconnect latency
   ( $L_{avg}$ )
3  $P \leftarrow$  Set of non-overlapping classes
4  $\Lambda \leftarrow 0$ 
5 for  $p \in P$  do
6    $G_p \leftarrow$  Set of arbiters for  $p$ 
7   for  $g_p \in G_p$  do
8      $L_p \leftarrow 0$ ;  $\Lambda_p \leftarrow 0$ 
9     Transform each hybrid priority queue into
      tandem queues as in Fig.3
10     $J \leftarrow$  Number of classes in  $g_p$ 
11    for  $j = 1 : J$  do
12      while  $\hat{\lambda}_j, \hat{C}_{aj}$  are not converged do
13        /* forward pass */
14        Obtain the modified injection process
          ( $\hat{\lambda}_j, \hat{C}_{aj}$ ) using the technique
          described in Section IV-B
15        Compute the average waiting time ( $W_j$ )
          for each class
16        /* backward pass */
17        Obtain the probability of full
18      end
19       $W_j = W_j + zero\_load\_latency(j)$ 
20       $L_p = L_p + \lambda_j W_j$ ;  $\Lambda_p = \Lambda_p + \lambda_j$ 
21    end
22     $\Lambda = \Lambda + \Lambda_p$ ;  $L_{tot} = L_{tot} + \frac{L_p}{\Lambda_p}$ 
23  end
24 end
25  $L_{avg} = \frac{L_{tot}}{\Lambda}$ 

```

---

The co-efficients  $a_0$  and  $a_1$  in the correction term  $a_0N + a_1$  is added in such a way that we can put the upper limit on  $n_j$  as  $N\hat{t}_j$ , the worst case latency each packet faces.

### D. End-to-End Latency Estimation

In this section, we describe our proposed algorithm to obtain the end-to-end latency for NoCs with hybrid arbitration. The input to the algorithm is the queuing network of which the NoC is comprised of, service process at each arbiter, arrival statistics of each class in the network and the size of each queue in the network. The queuing network consists of a series of tandem queues, with each set of queues being arbitrated via hybrid arbitration. First, we extract the set of non-overlapping source-destination pairs (class) in the network. For each element in the set, we find out the set of hybrid arbiters. We then transform the hybrid arbitration using our proposed technique shown in Figure 3. Then, for each set of arbiters we evaluate the classes as shown in line 10 of the algorithm. The end-to-end latency estimation for each class is an iterative process. Each iteration consists of two stages – forward pass and the backward pass. In the forward pass, the effective injection rate

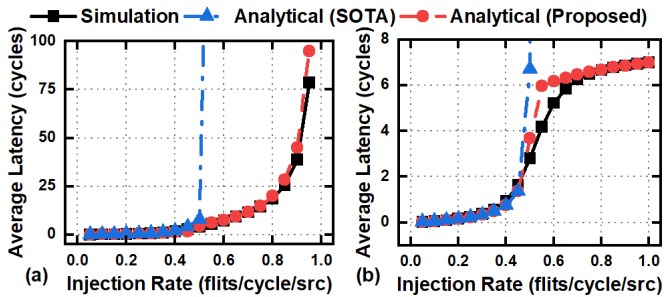


Fig. 4. Average Latency for (a) lower priority queue in priority arbitration and (b) round-robin arbitration of two queues with buffer size 4.

and the squared coefficient of variation of inter-arrival time of each queue with different arbitration is computed following the technique described in Section IV-B. Then, with the help of the effective arrival statistics, we apply the technique described in Section IV-C to obtain the waiting time of each queue in the presence of hybrid arbitration under high traffic injection. In the backward pass, the effect of back-pressure from the downstream queues is captured. To this end, the probability that each queue is full is computed using [16], with the help of the waiting time computed in the forward pass. The injection rate to each queue is further modified through the probability of the queues being full. The iteration is stopped when the percentage difference of the effective injection rate between two consecutive iterations is less than a user-defined threshold. After we obtain the waiting time ( $W_j$ ) of each class- $j$ , we add the zero load latency to the waiting time. Finally, we obtain the average latency of the entire NoC by taking the weighted average of the waiting time of all classes.

## V. EXPERIMENTAL EVALUATION

### A. Experimental Setup

We analyze our proposed analytical model in detail with NoCs having hybrid arbitration, different buffer sizes, and a wide range of injection rates. We vary the injection rate from a very low value to an injection rate where the NoC is under high traffic injection. The high injection rate can be achieved under several recently emerged compute and memory-intensive applications. The end-to-end average latency of the packets in the NoC is obtained from the proposed analytical model. We show the latency comparison with respect to BookSim [5] as well as a cycle-accurate simulator widely used in the industry [9]. We modify the BookSim to incorporate hybrid arbitration. The modified BookSim repository has been made public (link to repo). The source code of the analytical model will be made available publicly upon acceptance of the manuscript. The simulations are executed for 1 million cycles with a warm-up of 10,000 cycles to obtain steady-state latency. We note that there exists no analytical modeling technique for NoCs with hybrid arbitration while having high traffic injection. We compare our proposed analytical model with the one proposed in [1, 4] (denoted as ‘SOTA’ in the figures).

### B. Evaluation of Analytical Model under High Injection

We first evaluate our proposed analytical model with priority arbitration (shown in Figure 1(a)) and round-robin arbitra-

tion (shown in Figure 1(b)) under high traffic injection. We consider only two classes in these cases. Figure 4 shows the comparison in average packet latency between cycle-accurate simulation and our proposed analytical model for (a) priority and (b) round-robin arbitration with a buffer size of 4. With high injection rate and low buffer size, there is high congestion in the NoC, making latency estimation challenging. Our proposed analytical model always achieves less than 5% error for all injection rates for priority arbitration as seen in Figure 4(a). State-of-the-art analytical (‘SOTA’) model [1] is able to estimate the average latency accurately till the injection rate where the queues reach saturation. However, they overestimate the average latency under high traffic injection. In our proposed analytical model, we accurately estimate the effective injection rate to the queues under high traffic injection which results in accurate estimation of average latency.

Similar to the results for priority arbitration, we achieve high accuracy for round-robin arbitration as shown in Figure 4(b). We observe that, our proposed analytical model incurs only 5% error on average across all injection rates. However, state-of-the-art analytical model [4] incurs more than 100% error under high traffic injection.

### C. Evaluation with Different NoC Sizes and Buffer Sizes

In this section, we compare our proposed analytical model with respect to a widely used cycle accurate simulator – BookSim [5]. Figure 5 shows detailed results for  $8 \times 8$  mesh NoC having hybrid arbitration with four different buffer sizes. The average latency values are normalized with respect to the highest latency seen in simulation for that particular buffer size. It is seen that for all four buffer sizes, both SOTA and our proposed model estimate the average latency accurately with respect to simulation at low injection rate. However, at high injection rate, SOTA highly overestimates the latency; while our proposed analytical model estimates the latency accurately.

Table III shows the estimation error incurred by SOTA as well as our proposed analytical model for different injection rates, buffer sizes and NoC dimensions. Specifically, we are showing the results at high injection rates here. We observe that our proposed analytical model consistently estimates the average latency with more than 90% accuracy (except one case). In contrast, SOTA model incurs more than 100% estimation error in most of the cases when the injection rate is high. Since we are able to accurately compute the modified injection rate for hybrid arbitration under high injection, our proposed analytical model is accurate in high injection region irrespective of NoC sizes and buffer sizes.

### D. Evaluation on Industrial NoC with Hybrid Arbitration

Finally, we evaluate the proposed analytical model for an industrial NoC [9]. The NoC connects up to 84 nodes; where each node represents either a computing core, cache or memory controller. The NoC adheres a hybrid topology and consists of hybrid arbitration [9].

We choose a mix of workloads from SPEC2007 benchmark suite [20] to exhibit high traffic injection. For example, the

TABLE III  
COMPARISON OF AVERAGE ERROR BETWEEN SIMULATION AND ANALYTICAL MODEL. ‘E’ SIGNIFIES ERROR MORE THAN 100%.

Buffer Size	(a) 4×4								(b) 6×6							
	4		6		10		32		4		6		10		32	
Injection rate (packets/cycle/src)	0.7	0.8	0.7	0.8	0.7	0.8	0.7	0.8	0.7	0.8	0.7	0.8	0.7	0.8	0.7	0.8
SOTA Error(MAPE)% [1, 4]	E	E	17.07	22.25	E	E	E	E	E	E	E	E	E	E	E	E
Proposed Error(MAPE)%	7.62	7.10	0.61	4.82	3.67	1.20	3.54	11.25	5.86	5.11	1.82	7.82	1.53	1.65	0.95	1.45

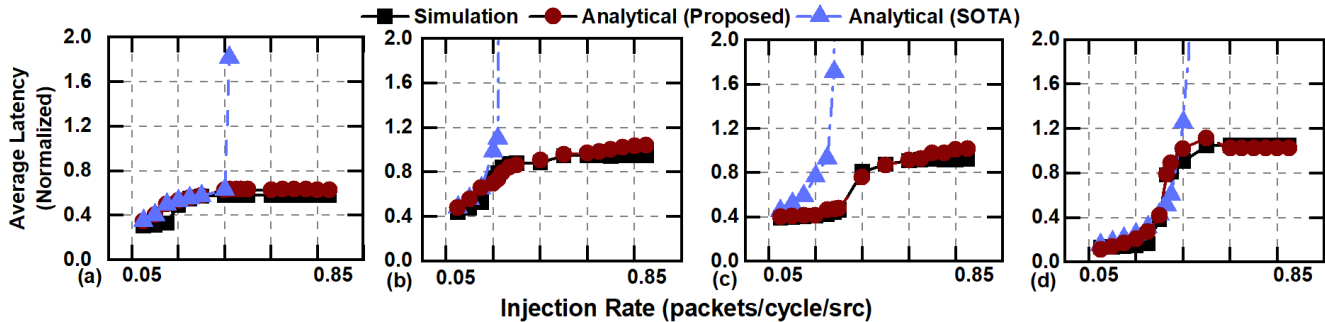


Fig. 5. Comparison of average latency between simulation and analytical model for  $8 \times 8$  mesh NoC with buffer size of (a) 4, (b) 6, (c) 10 and (d) 32.

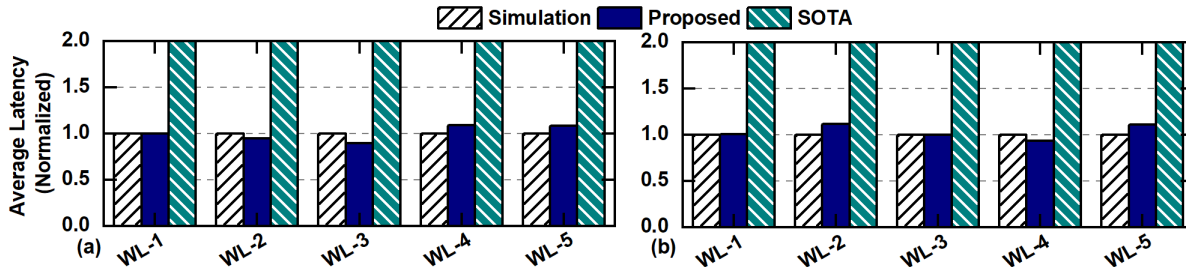


Fig. 6. Average packet latency for an NoC with hybrid arbitration with (a) buffer size of 2 and (b) buffer size of 4 while executing real applications having high traffic injection.

workload mix ‘WL-1’ denotes a mixture of libquantum and mcf workloads. The injection rates of the workloads vary from 0.4 to 0.9 packets/cycle and they show high burstiness. The burstiness of the traffic is taken care by the second order moment of the arrival rate in the analytical model. We simulate the real applications on the real system and obtain the average packet latency. The arrival rate statistics are computed from the application trace which is the input to our proposed analytical model to obtain the average packet latency. The comparison between the latency obtained through simulation and analytical model is shown in Figure 6(a) and Figure 6(b) for buffer sizes 2 and 4 respectively. The buffer sizes are chosen according to the real NoC. We observe that our proposed analytical model incurs only 7% and 6% on average across all applications with respect to the cycle-accurate simulation for buffer sizes 4 and 2 respectively. The state-of-the-art analytical model overestimates the latency by more than 100% for all these applications. Therefore, our analytical modeling technique is accurate for real applications executing on an industrial NoC with hybrid arbitration.

#### E. Speed-up Achieved Compared to Cycle-Accurate Simulator

The proposed analytical model and the simulation are implemented in C++ to perform a fair comparison in execution time. The NoC is simulated for a large number of cycles, long enough to reach the steady-state condition. We observe that our proposed analytical model achieves at least 3 orders of

TABLE IV  
SPEED-UP ACHIEVED FOR DIFFERENT NOC SIZES.

Mesh Size	4×4	6×6	8×8
Speed-up	1101.50	1453.55	1411.75

magnitude speed-up with respect to cycle-accurate simulator. The speed-up achieved for different NoC sizes are reported in Table IV. Hence, our proposed analytical model is accurate to estimate the performance of industrial NoCs with hybrid arbitration and provides significant speed-up with respect to the cycle-accurate simulator.

#### VI. CONCLUSION

This work proposes an analytical performance model for industrial NoCs with hybrid arbitration and high traffic injection. We propose a novel transformation to evaluate the performance of queuing systems with hybrid arbitration as well as a technique to compute the effective injection rate at high injection region. The proposed technique is evaluated against different NoC configurations across a wide range of injection rates showing less than 7% error in different scenarios with industrial NoC. The analytical model also achieves up to 3 orders of magnitude speed-up with respect to the cycle-accurate simulator.

#### ACKNOWLEDGMENT

This work is supported in part by research grant from Intel Corporation (FA/OTHER-23-0007) and Walmart Center for Tech Excellence (CSR Grant WMGT-23-0001) at IISc.

## REFERENCES

- [1] S. Y. Narayana *et al.*, “Fast Analysis Using Finite Queuing Model for Multilayer NoCs,” *IEEE Design & Test*, vol. 40, no. 6, pp. 112–124, 2023.
- [2] U. Y. Ogras *et al.*, “An Analytical Approach for Network-on-Chip Performance Analysis,” *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 29, no. 12, pp. 2001–2013, 2010.
- [3] A. E. Kiasari *et al.*, “An Analytical Latency Model for Networks-on-Chip,” *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 21, no. 1, pp. 113–123, 2012.
- [4] S. K. Mandal *et al.*, “Fast performance analysis for nocs with weighted round-robin arbitration and finite buffers,” *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 31, no. 5, pp. 670–683, 2023.
- [5] N. Jiang *et al.*, “A Detailed and Flexible Cycle-accurate Network-on-Chip Simulator,” in *2013 IEEE international symposium on performance analysis of systems and software (ISPASS)*, 2013, pp. 86–96.
- [6] N. Binkert *et al.*, “The Gem5 Simulator,” *ACM SIGARCH computer architecture news*, vol. 39, no. 2, pp. 1–7, 2011.
- [7] V. Catania *et al.*, “Noxim: An Open, Extensible and Cycle-accurate Network-on-Chip Simulator,” in *2015 IEEE 26th international conference on application-specific systems, architectures and processors (ASAP)*. IEEE, 2015, pp. 162–163.
- [8] A. Vaswani, “Attention is All you Need,” *Advances in Neural Information Processing Systems*, 2017.
- [9] semiWiki. Netspeed Systems. <https://semiwiki.com/x-subscriber/netspeed-systems/4825-netspeed-noc-ip-or-architectural-synthesis-company/>, accessed 11 Nov 2024.
- [10] “Netspeed orion: A new approach to design on-chip interconnects,” in *2013 IEEE Hot Chips 25 Symposium (HCS)*, 2013, pp. 1–19.
- [11] E. Fischer *et al.*, “An Accurate and Scalable Analytic Model for Round-robin Arbitration in Network-on-Chip,” in *Proc. of Intl. Symp. on Networks-on-Chip (NoCS)*, 2013, pp. 1–8.
- [12] O. J. Boxma *et al.*, “Waiting-time Approximations in Multi-queue Systems with Cyclic Service,” *Performance Evaluation*, vol. 7, no. 1, pp. 59–70, 1987.
- [13] Z.-L. Qian *et al.*, “A Support Vector Regression (SVR)-based Latency Model for Network-on-Chip (NoC) Architectures,” *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 35, no. 3, pp. 471–484, 2015.
- [14] J. Walraevens, “Discrete-time queueing models with priorities,” Ph.D. dissertation, 01 2004.
- [15] X. Jin *et al.*, “Modelling and analysis of priority queueing systems with multi-class self-similar network traffic: A novel and efficient queue-decomposition approach,” *IEEE Transactions on Communications*, vol. 57, no. 5, pp. 1444–1452, 2009.
- [16] G. Bolch *et al.*, *Queueing Networks and Markov chains: Modeling and Performance Evaluation with Computer Science Applications*. John Wiley & Sons, 2006.
- [17] S. K. Mandal, R. Ayoub, M. Kishinevsky, M. M. Islam, and U. Y. Ogras, “Analytical performance modeling of nocs under priority arbitration and bursty traffic,” *IEEE Embedded Systems Letters*, vol. 13, no. 3, pp. 98–101, 2020.
- [18] D. D. Kouvatsos, “Maximum Entropy and the G/G/1/N Queue,” *Acta Informatica*, vol. 23, pp. 545–565, 1986.
- [19] D. Bertsekas *et al.*, *Data Networks*. Athena Scientific, 2021.
- [20] J. Bucek *et al.*, “Spec cpu2017: Next-generation compute benchmark.” New York, NY, USA: Association for Computing Machinery, 2018. [Online]. Available: <https://doi.org/10.1145/3185768.3185771>