Latency-Aware QoS Optimization of XY-YX Routing in NoCs via Analytical Latency Estimation

Jongwon Oh
Seoul National University of Science and Technology
Seoul, Republic of Korea
ohjongwon@seoultech.ac.kr

Seongmo An
Seoul National University of Science and Technology
Seoul, Republic of Korea
ahnseongmo@seoultech.ac.kr

Jinyoung Shin
Seoul National University of Science and Technology
Seoul, Republic of Korea
shinjinyoung@seoultech.ac.kr

Seung Eun Lee*
Seoul National University of Science and Technology
Seoul, Republic of Korea
seung.lee@seoultech.ac.kr

Abstract

Efficient communication in on-chip networks (NoCs) is essential for high-performance and energy-efficient many-core systems. As network size and workloads increase, bursty traffic and contention significantly impact system latency. While XY-YX routing is a simple and deadlock-free deterministic method, it often struggles to deliver optimal latency under bursty traffic conditions. This paper introduces an optimization scheme for XY-YX routing based on analytical latency estimation in mesh-based NoCs. Unlike prior node- or link-centric approaches, our method directly estimates the end-to-end latency of routing sequences in the presence of bursty traffic, allowing for more effective quality-of-service (QoS) optimization. We present an analytical model that captures XY-YX path characteristics as well as the impact of bursty traffic on link utilization and contention. By integrating this analytical latency model into routing decisions, our approach is able to select routing sequences that minimize packet delay under traffic loads. The proposed method achieves latency estimation with over 93% accuracy compared to RTL simulation while requiring significantly lower computational overhead. It also supports exhaustive routing assignment evaluation across large-scale traffic patterns within practical runtime. This enables fast optimization of XY-YX routing by selecting latency-minimizing path assignments, supporting balanced trade-offs among latency, fairness, and contention.

CCS Concepts

• **Hardware → Network on chip**; • **Networks → Network performance modeling**.

Keywords

modeling and prediction, network-on-chip (NoC), quality-of-service (QoS), discrete-time queuing theory

1 Introduction

As modern system-on-chip (SoC) architectures continue to scale in complexity, integrating dozens to hundreds of heterogeneous processing cores and memory units on a single die, the importance of efficient on-chip communication has steadily increased. Network-on-chip (NoC) has emerged as a widely adopted interconnection solution to address the scalability, modularity, and performance requirements of such systems [3, 16, 18]. In these environments, routing strategies contribute to determining how data packets move through the network and have a noticeable effect on system-level aspects such as communication latency, throughput, and energy efficiency [21]. To support fast and reliable communication under shared bandwidth constraints, NoC designs increasingly require not only architectural scalability, but also adaptable routing methods.

One particular challenging situation arises when bursty traffic occurs—that is, when multiple packets are injected into the network fabric simultaneously from different nodes [8, 14, 17]. Such traffic patterns, which are common in various parallel processing workloads, lead to contention and temporary bottlenecks, potentially increasing overall response time and affecting system performance. To maintain efficient communication under these conditions, routing approaches that mitigate the impact of contention and provide a more balanced distribution of packet flows are crucial.

Prior studies have been proposed to address traffic contention in NoCs, including both adaptive routing protocols and reconfigurable network architectures [4, 20]. While reconfigurable topologies offer the flexibility to dynamically reshape communication paths, they often require additional hardware resources and introduce control complexity and power overhead. In scenarios where maintaining a fixed network architecture is preferable for reasons of simplicity or cost, it becomes meaningful to consider alternatives that focus on routing-level coordination rather than architectural changes.

In this context, XY-YX routing provides a simple and deterministic method that allows packets to follow either an XY (x-axis first, then y-axis) or YX (y-axis first, then x-axis) path through a 2D mesh network [7]. Although XY-YX routing is deadlock-free and hardware-efficient, its effectiveness depends on how the two path types are assigned across packets. If a number of packets choose the same routing direction, localized contention occurs, reducing the overall benefit of having multiple routing options. This observation suggests that systematic assignment of XY and YX paths is able to reduce hotspot contention and improve load balancing [1].

*Corresponding author.

CAMS'25, co-located with MICRO 2025, Seoul, Korea

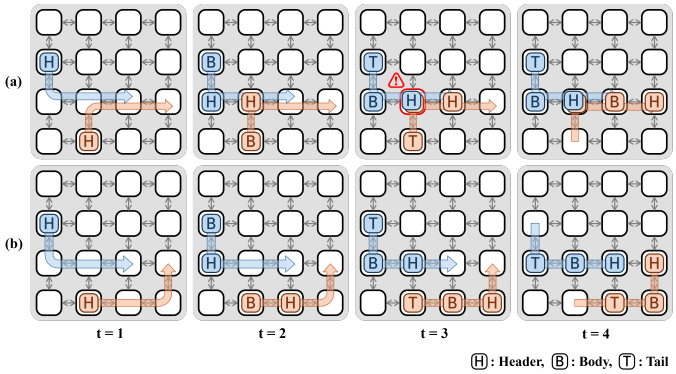

Figure 1: Time-step demonstration of packet movements in a 2D mesh NoC. (a) shows a case where two packets encounter contention at a shared node. (b) shows how contention is avoided by adjusting routing paths.

This work explores how such routing assignments are optimized by modeling their latency impact analytically. We present a packet-level routing optimization framework that estimates the total communication delay under different XY-YX combinations. By leveraging queuing theory to account for contention and link utilization, the proposed method predicts end-to-end latency across multiple packets and identifies routing combinations that offer improved traffic balance. Unlike purely heuristic or locally adaptive methods, our approach considers interactions across the entire network, enabling a more informed assignment process. Fig. 1 provides a conceptual example illustrating how different routing combinations influence contention patterns. In one case, two packets compete for the same intermediate node, resulting in contention. In the other, the packets are assigned alternate routing paths, avoiding the overlap and improving overall flow.

To evaluate the accuracy and practicality of our latency model, we implemented a verilog-based cycle-accurate simulation of an XY-YX router operating on a 2D mesh topology. Simulation results showed that the estimated latency values closely align with actual simulation outcomes, indicating that the proposed model captures traffic dynamics with reasonable accuracy. Moreover, the model's simplicity allows it to be used in design-time scenarios where routing combinations have to be evaluated quickly.

The main contributions of this paper are summarized as follows:

- We present a latency estimation model for XY-YX routing that enables optimal assignment selection under bursty traffic conditions.
- The proposed framework is lightweight and capable of evaluating and optimizing latency across dozens of packet combinations.
- We validated the effectiveness of the model through hardware implementation and function-level simulations.

These contributions are expected to provide practical guidance for the design of NoC routing algorithms and traffic scheduling strategies, especially in the context of latency-sensitive multicore and heterogeneous systems.

2 Related Work

The performance analysis of NoCs has been explored through a range of modeling approaches, particularly those that aim to estimate latency under various routing and traffic conditions [6, 8, 10–12]. Among these, lightweight analytical models have received attention as they offer practical alternatives to detailed simulations.

One such model is presented by [8], which proposed an analytical framework for NoCs operating under priority arbitration and bursty traffic scenarios. Their approach uses a generalized geometric distribution to model bursty traffic injection and applies maximum entropy-based techniques to solve per-router queuing systems. The framework reports latency estimates with relatively low modeling errors compared to simulation, while also maintaining low computational complexity. Although the model addresses realistic traffic behavior, it focuses on priority-aware scheduling rather than routing-level path selection, which limits its applicability in static mesh environments where path assignment is fixed at injection time.

In another line of work, [6] introduced an analytical latency model for deterministic routing in wormhole-switched NoCs. This model estimates average latency based on network topology and traffic load, and supports various routing algorithms, including XY and its variants. While the method provides general insights into NoC behavior under different configurations, it primarily targets average-case latency and does not account for concurrent multi-packet scenarios or route assignment flexibility, which are relevant to routing strategy selection under bursty injection.

Complementing these studies, [12] investigated routing algorithms designed for specific applications in NoC systems. Their work examines how static routing methods such as XY and YX manage traffic more effectively, particularly when routing decisions are tailored to communication patterns. Although their approach is heuristic in nature and does not rely on analytical modeling, it highlights the potential of route assignment strategies to influence network performance, especially in systems with constrained or deterministic routing schemes.

These prior works have addressed performance modeling from various perspectives, including queuing analysis, arbitration mechanisms, and application-driven routing decisions. However, they have not explicitly considered the problem of selecting optimal routing directions (e.g., XY or YX) on a per-packet basis to minimize overall network latency. In this context, the approach proposed in this paper—an analytical estimation framework designed to evaluate total latency for different combinations of XY and YX routing in static 2D mesh topologies—is positioned as a complementary method to existing models, offering a lightweight alternative for routing strategy evaluation. The framework aims to support efficient exploration of routing configurations while maintaining alignment with functional simulation results.

3 Background

3.1 XY-YX Router Design

In this work, we focus on modeling and optimizing packet latency under the XY-YX routing method in 2D mesh-based NoCs. The XY-YX routing approach allows each packet to follow either the

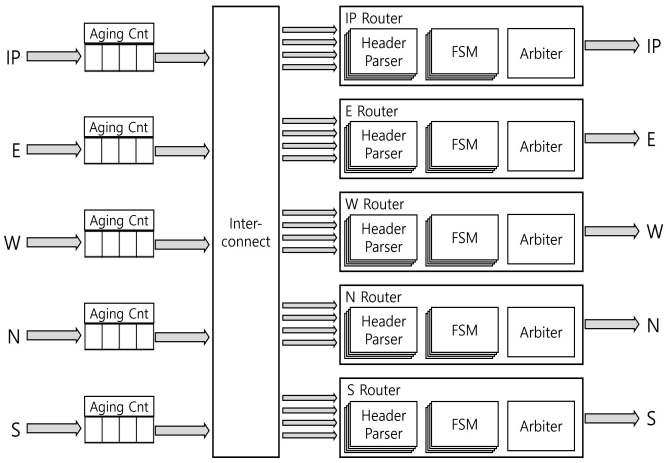

Figure 2: Overall architecture of the XY-YX router, featuring per-direction arbitration across five input/output ports.

conventional XY or YX path from source to destination. These routing decisions are determined at injection time and remain fixed throughout the packet's traversal. Since all routing paths are deterministic and do not involve adaptive decision or backtracking, the XY-YX scheme inherently guarantees deadlock freedom—a critical requirement for stable on-chip communication systems.

To support this routing approach in hardware, we implemented a router capable of directing packets based on their assigned XY or YX routing mode. The overall architecture of this XY-YX router is illustrated in Fig. 2. The router is designed for a standard 2D mesh topology and consists of five ports: one local IP port and four directional ports (East, West, North, and South). Each input direction is associated with an internal buffer, and routing decisions are made independently per output direction.

The router adheres to the wormhole flow control protocol, in which a packet occupies routing resources along its path until it has been completely forwarded. This mechanism allows for compact buffer usage, but also makes the system susceptible to contention when multiple packets compete for the same output direction. To mitigate this, the XY-YX router architecture incorporates direction-specific arbiters for each output port, enabling independent arbitration and reducing the likelihood of contention across different routing paths.

Contention is further minimized through the decoupling of arbitration per output direction: packets requesting different output directions are handled independently and proceed concurrently. Contention occurs only when multiple input packets request access to the same output port. In such cases, a fixed-priority arbitration scheme is employed to resolve conflicts. The fixed-priority order is determined statically in a clockwise fashion around the router, ensuring consistent and predictable resolution policies [13].

However, fixed-priority arbitration has the potential to cause starvation in the presence of persistent contention, particularly for directions with lower assigned priorities. To address this issue, the design integrates an aging mechanism into the input buffers. Each buffer maintains an aging counter, which increases over time and elevates the packet's priority. This dynamic adjustment prevents long-term blocking and promotes fairness among all incoming directions.

Overall, this router architecture is tailored to support the evaluation and implementation of XY-YX routing strategies under bursty traffic conditions. By explicitly separating arbitration logic and introducing starvation prevention mechanisms, the design provides a robust basis for evaluating contention behavior under varying XY-YX routing combinations [5].

3.2 Discrete-Time Queuing Theory for Latency Estimation

To analytically estimate the latency of multiple packets under simultaneous injection, our study adopts a modeling approach based on discrete-time queuing theory. In NoC simulation and performance analysis, time-based queuing models are important, particularly for understanding contention phenomena across nodes and channels [15]. When both packet arrivals and services are governed by discrete time event, queuing dynamics are accurately captured with a discrete-event framework.

The discrete-time queuing theory introduced by Meisling assumes a system where all events—packet arrivals and services—occur at fixed time intervals [9]. Customers (in our case, packets) arrive according to a discrete binomial distribution, and the service times are represented by a probability distribution or fixed latency. Under these assumptions, the average queue length, waiting time, and service delay are characterized as a function of the system's traffic density ρ, arrival rate λ and service time distribution. The traffic density ρ is determined by the expected service time $E(s)$ and the arrival rate λ, $\rho = \lambda E(s)$. The expected service time is expressed as (1) where C_k represents the probability that a service takes $k\Delta t$ time. This reflects the statistical behavior of packet servicing within the router modules and allows the derivation of expected latency under various traffic intensities.

$$E(s) = \sum_{k=0}^{\infty} C_k \cdot k\Delta t \tag{1}$$

Furthermore, for a given service time distribution and traffic density ρ in a stable system ($\rho < 1$), the expected waiting time $E(w)$ is approximated as (2).

$$E(w) = \frac{\lambda E[s(s - \Delta t)]}{2(1 - \rho)} \tag{2}$$

These formulations serve as the analytical foundation for our lightweight latency estimation framework. Unlike cycle-accurate simulations, which are computationally intensive, this model allows rapid evaluation of packet completion time under varying XY-YX routing assignments [2].

Importantly, this model is well aligned with the synchronous operation of NoC routers and channels, which operate on a global clock. The discrete-time nature of the model allows packet movements, buffer updates, and contention events to be modeled at the granularity of a single clock cycle. In this work, we independently model each node's arrival and service process to evaluate how different routing combinations impact the total latency. The

derived statistics—based on queuing behavior and packet traversal time—are then used to estimate the average latency, hop-level contention, and system-level completion time across all packets.

4 Methodology

4.1 Analytical Latency Estimation Model

The simulation framework operates based on a discrete-time event-driven model, where all system activities occur only at specific time points. Within a single time tick, packets can be injected simultaneously from multiple source nodes; however, each source node is restricted to injecting only one packet at a time.

The target system is a 2D mesh-based NoC, where each node is connected to four neighboring nodes through directional channels, and deterministic XY-YX routing is applied. Each node is assumed to have a single-server output queue for each direction, and the queue has infinite buffer capacity (infinite buffer size assumption). Thus, no packet drop occurs under contention, and packets are served in a FIFO manner based on their arrival order.

The proposed method performs path-level latency analysis by summing hop-based transmission delay and contention-induced waiting time for each communication path. Contention may occur between paths that share the same output direction at same node, which increases queue length and introduces additional delay. Such contention is dynamically detected and quantitatively analyzed based on discrete-time queuing theory. For each path, the expected waiting time $E(w)$ is calculated and reflected in the total latency, enabling accurate evaluation of how internal NoC contention affects communication latency. This lightweight latency estimation approach is applicable under various packet combinations and traffic conditions, while also enabling assignment exploration and traffic routing analysis. The parameters used in the modeling are summarized in Table 1.

4.1.1 Path-Level Latency Model. In this study, NoC communication is modeled on a path basis according to a deterministic routing scheme, and the latency components of each path are quantitatively analyzed. The simulation framework estimates the total latency by summing the transmission delay and the queueing delay caused by contention along the given path. The total latency for path i is expressed as follows:

$$L_i = D_i^{S\to D} + \sum_{n\in C_i} E(w)_i^n \tag{3}$$

where D_i denotes the transmission delay from source node S to destination node D, and $E(w)_i$ represents the expected waiting time caused by contention. Each packet is composed of multiple flits and is transmitted sequentially through intermediate nodes following the wormhole flow control. The header flit reaches the destination first, followed by the remaining $N_i - 1$ flits, which occupy the channel sequentially [19]. As each flit is transmitted only after the preceding flit advances, the minimum transmission delay is given by:

$$D_i = (N_i + h_i - 1) \cdot t_r \tag{4}$$

where h_i is the number of hops in path i, N_i is the number of flits in the packet, and t_r is the transmission time of a flit between adjacent nodes. The estimated waiting time $E(w)_i$, which corresponds to the queueing delay in path i, is calculated based on discrete-time queuing theory, considering contention at shared output nodes along path i.

Table 1: Parameter Notation

Symbol	Description
L	Average packet latency in the network *(cycles)*
	Path-level latency parameters
t_r	Time spent for transmitting a flit between adjacent nodes *(cycles)*
N_i	Size of packet in path i *(flits)*
K_i	Number of packet in path i *(packets)*
h_i	Hop count of path i *(hops)*
L_i	Packet latency on path i *(cycles)*
C_i	Set of contending paths for path i *(nodes)*
$D_i^{S\to D}$	Transmission latency from S to D *(cycles)*
$E(w)_i$	Estimated waiting time in path i *(cycles)*
	Node-level latency parameters
$P_{i,j}^n$	Probability to contention at node n between path i and j
λ_i	Packet arrival rate in path i *(packets/cycle)*
$E(s)_i^n$	Estimated service time at node n in path i *(cycles)*
ρ_i^n	Traffic density at node n in path i
$E(w)_i^n$	Estimated waiting time at node n in path i *(cycles)*

4.1.2 Node-Level Latency Model. The proposed simulation framework detects contention between paths and quantitatively estimates the resulting expected waiting time. Contention arises when two or more paths attempt to access the same output port at a shared node simultaneously, which becomes a major source of queuing delay. The degree of such contention-induced delay is quantified using the contention probability $P_{i,j}^n$, which reflects the likelihood that path j gains priority in using the output port at a contention node shared with path i.

The expected waiting time $E(w)_i$, for path i is estimated with discrete-time queuing theory, based on the contention probabilities $P_{i,j}^n$ of all paths j in the contention set C_i. The effective arrival rate is calculated using the following equation:

$$\lambda_i = \frac{K_i + \sum_{(j,n)\in C_i} K_j \cdot P_{i,j}^n}{D_i^{S\to D} + \sum_{(j,n)\in C_i} N_j \cdot t_r \cdot P_{i,j}^n} \tag{5}$$

In (5), the numerator represents the total traffic that affect path i, which is computed as the sum of its own packet count and the weighted sum of packets from contending paths, scaled by the corresponding contention probabilities. The denominator consists of the transmission delay $D_i^{S\to D}$ of path i and the aggregate delay contributions from the contending paths that may occupy the shared nodes. Eq. (5), therefore, estimates the effective arrival rate λ_i, for path i under contention.

With the traffic density derived from this λ_i, the expected waiting time $E(w)_i$, is further estimated as follows:

$$\rho_i = \lambda_i E(s)_i = \lambda_i \sum_{(j,n)\in C_i} P_{i,j}^n \cdot N_j t_r \tag{6}$$

$$E(w)_i = \frac{\rho_i \cdot E(s)_i}{2(1-\rho_i)} \tag{7}$$

Through this modeling approach, the simulation framework quantitatively captures the delay caused by contention at shared nodes and incorporates it into the latency analysis for each path. This allows lightweight yet effective latency estimation that accounts for contention impact, and remains robust under various packet combinations and traffic patterns.

4.2 XY-YX Routing Optimization

In the proposed framework, routing optimization is performed by exhaustively evaluating all possible combinations of XY and YX routing for each packet (given n packets, there are 2^n routing combinations). For every configuration, the average latency across all packets is estimated using the analytical model described in the previous subsection. The routing assignment that yields the lowest average latency is selected as the optimal XY-YX configuration.

While cycle-accurate simulations could be utilized to perform such exhaustive evaluations, the computational cost is prohibitive, especially as the number of packets increases. In contrast, our analytical model enables rapid latency estimation with significantly reduced overhead, allowing exploration of routing configurations. This lightweight approach makes it feasible to perform routing optimization even under large-scale traffic scenarios, enabling efficient design space exploration without sacrificing accuracy.

5 Experimental Evaluation

5.1 Validation of the Latency Estimation Model

To validate the accuracy and robustness of our latency estimation model, we conducted a series of experiments comparing the results produced by our simulator against RTL-level simulations under matching conditions. Specifically, we evaluated two mesh topologies—4×4 and 8×8—both configured with deterministic XY-YX routing and wormhole switching as described in Section 3. Across these topologies, we applied a range of packet injection rates and observed the corresponding average packet latency in both the estimation model and RTL simulation.

Fig. 3 and Fig. 4 show the latency comparison results for each topology. Despite the lightweight nature of our estimation model, it closely matches the RTL simulation output across all injection rates. In the 8×8 mesh, the average error in estimated latency was just 2.8%, while the 4×4 mesh showed a slightly higher but still low error of 6.3%, validating the estimation model's reliability even as network size and traffic scale increased.

In addition to global averages, we further analyzed the latency behavior on a per-path basis. For each topology, we compared the average latency of every source-destination path between the estimated model and the RTL simulation. This path-level analysis revealed that the estimation model not only captures global trends, but also accurately reflects local congestion effects and contention-specific delay patterns that vary across individual paths. Fig. 5 and Fig. 6 illustrates this comparison, showing a high correlation between the two methods for all measured paths in each topology.

These findings confirm that the proposed latency estimation framework offers a high-fidelity, low-overhead alternative to cycle-accurate simulation. It captures the essential characteristics of queuing delay and contention effects across both global and local network behaviors. The model's ability to provide fast and accurate latency estimates enables scalable routing and traffic evaluation even under high-dimensional NoC scenarios.

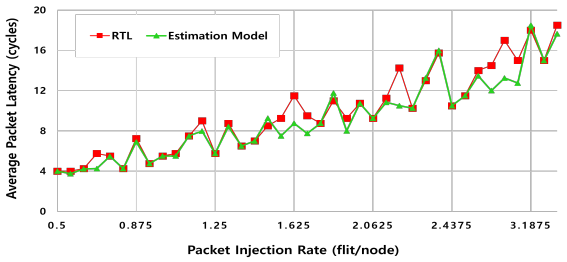

Figure 3: Average packet latency for a 4×4 mesh network.

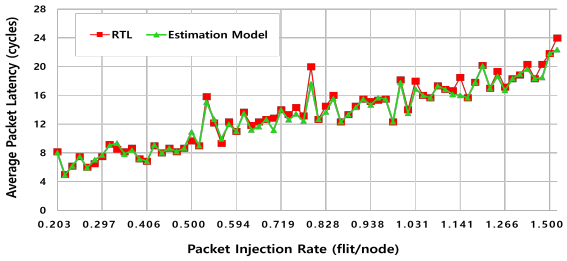

Figure 4: Average packet latency for an 8×8 mesh network.

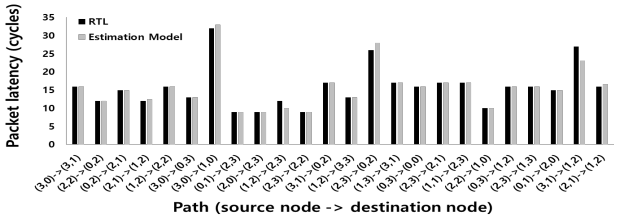

Figure 5: Packet latency for paths in a 4×4 mesh network.

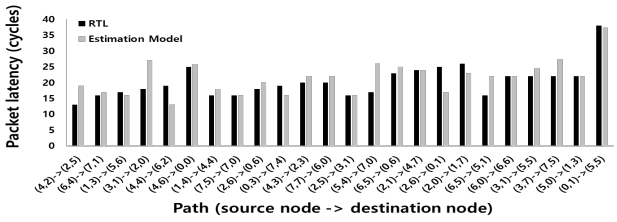

Figure 6: Packet latency for paths in an 8×8 mesh network.

Table 2: Evaluation of the XY-YX optimization

	Topo.	12×12 mesh										16×16 mesh									
	λ	0.3	0.4	0.6	0.8	0.9	1.1	1.2	1.3	1.4	1.5	0.4	0.6	0.7	0.8	0.9	1.0	1.1	1.2	1.4	1.5
Latency *(cycles)*	XY-only	10.5	13.6	15.2	17.4	18.6	18.9	21.0	22.2	22.3	24.5	21.1	25.5	26.0	29.2	31.9	32.7	34.2	43.9	44.0	56.4
	YX-only	10.6	13.4	15.9	16.5	20.2	17.5	21.7	22.7	24.2	23.1	20.8	25.8	25.4	33.1	31.2	35.7	34.9	41.4	47.6	48.7
	XY/YX-opt.	10.0	12.6	14.4	15.8	16.7	16.3	18.4	20.1	20.8	20.8	19.8	23.9	23.3	28.5	28.5	29.9	31.0	37.5	41.4	44.2

5.2 Evaluation of the XY-YX Routing Optimization

To evaluate the effectiveness of the proposed XY/YX routing optimization framework, we conducted a comprehensive set of experiments on two mesh topologies—12×12 and 16×16. For each topology, we applied the analytical latency model to all possible combinations of XY and YX path assignments, and selected the routing configuration that yielded the lowest average packet latency. The experiments were performed across a wide range of packet injection rates λ, and the results are summarized in Table 2.

As the injection rate increases, both XY-only and YX-only configurations exhibit a steep rise in average latency, due to their limited flexibility in distributing traffic load. In contrast, the proposed XY/YX-opt. strategy consistently achieves lower latency by balancing the routing directions across packets. For instance, in the 12×12 mesh with $\lambda = 1.4$, the average latency under XY-only and YX-only routing reaches 22.3 and 24.2 cycles, respectively, while the optimized configuration reduces it to 20.8 cycles. In the larger 16×16 mesh, the effect becomes more pronounced: $\lambda = 1.5$, the latency drops from 56.4 (XY-only) and 48.7 (YX-only) to 44.2 cycles under the optimal assignment. These results demonstrate the model's ability to effectively identify low-latency configurations under varying traffic conditions and topology scales.

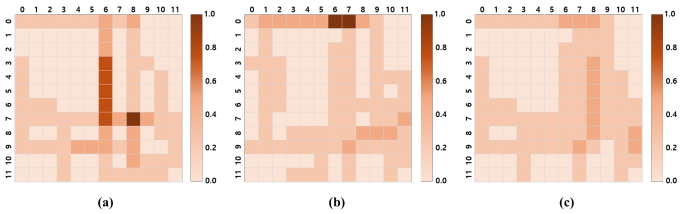

Figure 7: Node-level contention heatmaps for a 12×12 mesh NoC under (a) XY-only routing, (b) YX-only routing, and (c) optimized XY/YX routing configuration.

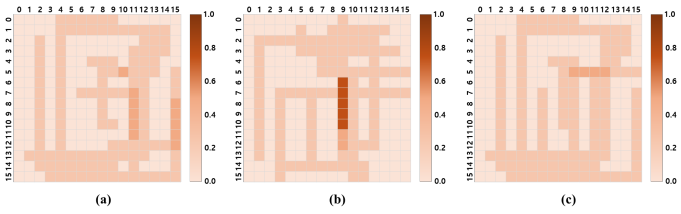

Figure 8: Node-level contention heatmaps for a 16×16 mesh NoC under (a) XY-only routing, (b) YX-only routing, and (c) optimized XY/YX routing configuration.

To further examine the cause of performance improvement, we visualized the node-level contention intensity for each routing scheme using heatmaps, as shown in Fig. 7 and Fig. 8. In both topologies, the XY-only and YX-only configurations result in strong contention concentration along specific rows or columns, reflecting low traffic distribution. In contrast, the XY/YX-optimal configuration spreads the traffic more evenly, resulting in less localized congestion and improved network efficiency.

These findings validate the practical benefit of the proposed estimation-based routing framework. Without relying on cycle-accurate simulation or complex adaptive routing, our method explores the design space rapidly and identifies balanced routing strategies that adapt to network size and traffic demand.

6 Conclusion

In this work, we proposed a lightweight analytical framework for evaluating and optimizing XY-YX routing assignments in mesh-based NoC environments. Motivated by the need to manage bursty traffic and contention without relying on hardware reconfiguration or cycle-accurate simulation, our method estimates the average packet latency based on discrete-time queuing theory and contention modeling.

The proposed model accurately captures the latency contributions from both hop traversal and queuing delay. Through extensive validation against RTL simulations, we demonstrated that the model achieves high accuracy with minimal overhead. Furthermore, the model supports rapid evaluation of routing combinations, making it suitable for design-time optimization.

By exhaustively analyzing routing assignments, the proposed XY/YX optimization strategy consistently reduced average packet latency and localized congestion compared to fixed-direction routing. These benefits were confirmed across different mesh sizes and injection rates, showing that even simple deterministic routing is able to be improved through contention-aware path assignment.

While the framework offers clear advantages in speed and scalability, its analytical nature may lead to minor underestimation of contention-induced delays, especially in high-congestion scenarios. This trade-off between lightweight modeling and accuracy should be considered when applying the method to latency-critical systems.

Overall, this work provides a practical and scalable solution for QoS-aware XY-YX routing optimization under bursty traffic. The analytical framework is expected to serve as a foundation for further research in NoC routing design, including adaptive schemes and traffic-aware mapping strategies.

References

[1] A. A. J. Al-Hchaimi, N. B. Sulaiman, M. A. B. Mustafa, M. N. B. Mohtar, S. L. B. M. Hassan, and Y. R. Muhsen. 2023. Evaluation Approach for Efficient Countermeasure Techniques Against Denial-of-Service Attack on MPSoC-Based IoT Using Multi-Criteria Decision-Making. *IEEE Access* 11 (2023), 89–106. doi:10.1109/ACCESS.2022.3232395

[2] R. V. De Liz Bomer, C. A. Zeferino, L. O. Seman, and V. R. Q. Leithardt. 2023. Worst-Case Communication Time Analysis for On-Chip Networks With Finite Buffers. *IEEE Access* 11 (2023), 25120–25131. doi:10.1109/ACCESS.2023.3255516

[3] Tom Glint, Manu Awasthi, and Joycee Mekie. 2024. CANSim: When to Utilize Synchronous and Asynchronous Routers in Large and Complex NoCs. In *2024 29th Asia and South Pacific Design Automation Conference (ASP-DAC)*. 1–6. doi:10.1109/ASP-DAC58780.2024.10473845

[4] P. Iff, M. Besta, M. Cavalcante, T. Fischer, L. Benini, and T. Hoefler. 2023. Sparse Hamming Graph: A Customizable Network-on-Chip Topology. In *2023 60th ACM/IEEE Design Automation Conference (DAC)*. San Francisco, CA, USA, 1–6. doi:10.1109/DAC56929.2023.10247754

[5] Y. S. Jeong and S. E. Lee. 2013. Deadlock-free XY-YX router for on-chip interconnection network. *IEICE Electronics Express* 10, 20 (2013), 1–5.

[6] A. E. Kiasari, Z. Lu, and A. Jantsch. 2013. An Analytical Latency Model for Networks-on-Chip. *IEEE Transactions on Very Large Scale Integration (VLSI) Systems* 21, 1 (Jan 2013), 113–123. doi:10.1109/TVLSI.2011.2178620

[7] S. M. Lee, E. N. R. Ko, Y. S. Jeong, and S. E. Lee. 2016. Design of a Deadlock-Free XY-YX Router for Network-on-Chip. In *Information Technology: New Generations*. Advances in Intelligent Systems and Computing, Vol. 448. Springer, Cham. doi:10.1007/978-3-319-32467-8_61

[8] S. K. Mandal, R. Ayoub, M. Kishinevsky, M. M. Islam, and U. Y. Ogras. 2021. Analytical Performance Modeling of NoCs under Priority Arbitration and Bursty Traffic. *IEEE Embedded Systems Letters* 13, 3 (Sept 2021), 98–101. doi:10.1109/LES.2020.3013003

[9] T. Meisling. 1958. Discrete-Time Queuing Theory. *Operations Research* 6, 1 (1958), 96–105. doi:10.1287/opre.6.1.96

[10] E. A. Monakhova, O. G. Monakhov, and A. Y. Romanov. 2023. Routing Algorithms in Optimal Degree Four Circulant Networks Based on Relative Addressing: Comparative Analysis for Networks-on-Chip. *IEEE Transactions on Network Science and Engineering* 10, 1 (Jan-Feb 2023), 413–425. doi:10.1109/TNSE.2022.3211985

[11] Y. R. Muhsen, N. A. Husin, M. B. Zolkepli, N. Manshor, and A. A. J. Al-Hchaimi. 2023. Evaluation of the Routing Algorithms for NoC-Based MPSoC: A Fuzzy Multi-Criteria Decision-Making Approach. *IEEE Access* 11 (2023), 102806–102827. doi:10.1109/ACCESS.2023.3310246

[12] M. Palesi, R. Holsmark, S. Kumar, and V. Catania. 2009. Application Specific Routing Algorithms for Networks on Chip. *IEEE Transactions on Parallel and Distributed Systems* 20, 3 (March 2009), 316–330. doi:10.1109/TPDS.2008.106

[13] P. Papaphilippou, K. Sano, B. A. Adhi, and W. Luk. 2023. Experimental Survey of FPGA-Based Monolithic Switches and a Novel Queue Balancer. *IEEE Transactions on Parallel and Distributed Systems* 34, 5 (May 2023), 1621–1634. doi:10.1109/TPDS.2023.3244589

[14] Zhiliang Qian, Paul Bogdan, Chi-Ying Tsui, and Radu Marculescu. 2016. Performance Evaluation of NoC-Based Multicore Systems: From Traffic Analysis to NoC Latency Modeling. *ACM Transactions on Design Automation of Electronic Systems* 21, 3 (July 2016), Article 52, 38 pages. doi:10.1145/2870633

[15] U. V. Rane, C. Panem, G. Abhyankar, and R. S. Gad. 2024. Network on Chip(NoC) Mesh Topology FPGA Verification: Real Time Operating System Emulation Framework. In *2024 Fourth International Conference on Advances in Electrical, Computing, Communication and Sustainable Technologies (ICAECT)*. Bhilai, India, 1–5. doi:10.1109/ICAECT60202.2024.10468944

[16] B. N. K. Reddy, M. Zia Ur Rahman, and A. Lay-Ekuakille. 2024. Enhancing Reliability and Energy Efficiency in Many-Core Processors Through Fault-Tolerant Network-on-Chip. *IEEE Transactions on Network and Service Management* 21, 5 (Oct 2024), 5049–5062. doi:10.1109/TNSM.2024.3394886

[17] V. Soteriou, Hangsheng Wang, and L. Peh. 2006. A Statistical Traffic Model for On-Chip Interconnection Networks. In *14th IEEE International Symposium on Modeling, Analysis, and Simulation*. Monterey, CA, USA, 104–116. doi:10.1109/MASCOTS.2006.9

[18] Haoyu Wang and Basel Halak. 2023. Hardware Trojan Detection and High-Precision Localization in NoC-based MPSoC using Machine learning. In *2023 28th Asia and South Pacific Design Automation Conference (ASP-DAC)*. 516–521.

[19] X. Xiang, P. Sigdel, and N.-F. Tzeng. 2020. Bufferless Network-on-Chips With Bridged Multiple Subnetworks for Deflection Reduction and Energy Savings. *IEEE Trans. Comput.* 69, 4 (April 2020), 577–590. doi:10.1109/TC.2019.2959307

[20] D. Xu, Y. Ouyang, W. Zhou, Z. Huang, H. Liang, and X. Wen. 2023. RMC_NoC: A Reliable On-Chip Network Architecture With Reconfigurable Multifunctional Channel. *IEEE Transactions on Very Large Scale Integration (VLSI) Systems* 31, 12 (Dec 2023), 2061–2074. doi:10.1109/TVLSI.2023.3321598

[21] Y. Xue et al. 2024. Automatic Generation and Optimization Framework of NoC-Based Neural Network Accelerator Through Reinforcement Learning. *IEEE Trans. Comput.* 73, 12 (Dec 2024), 2882–2896. doi:10.1109/TC.2024.3441822