RESEARCH ARTICLE

Asian Research Association

INTERNATIONAL RESEARCH JOURNAL OF
MULTIDISCIPLINARY TECHNOVATION

DOI: 10.54392/irjmt2629

Performance Evaluation of Network-On-Chip Routing Algorithms across Multiple Traffic Scenarios

Prajakta P. Dere [a], Shweta S. Ashtekar [a], Ekta Sarda [b], Bhushan S. Deore [c, *]

[a] Department of CE, D Y Patil University, Ramrao Adik Institute of Technology, Navi Mumbai – 400706, Maharashtra, India
[b] Department of CSE, D Y Patil University, Ramrao Adik Institute of Technology, Navi Mumbai – 400706, Maharashtra, India
[c] Department of ECE, D Y Patil University, Ramrao Adik Institute of Technology, Navi Mumbai – 400706, Maharashtra, India
* Corresponding Author Email: bhushan.deore@dypatil.edu

DOI: https://doi.org/10.54392/irjmt2629
Received: 07-11-2024; Revised: 26-02-2026; Accepted: 04-03-2026; Published: 16-03-2026

Abstract: The significance of effective on-chip communication has increased due to the increasing integration of computing cores in contemporary systems. Since routing techniques affect latency, throughput, and power consumption while reducing congestion and deadlock, they are essential to Network-on-Chip (NoC) performance. While adaptive and hybrid algorithms like Q-learning (Q), Path-based Randomized Oblivious Minimal (PROM), and Dynamic Adaptive Deterministic (DyAD) promise greater adaptability, deterministic techniques like XY provide simplicity but lack flexibility. The deterministic, adaptive, and hybrid routing algorithms in NoC are assessed in this study under bursty and constant bit rate (CBR) traffic. By combining delay, throughput, and power, a composite Performance Metric (PM) is used to measure routing efficiency. According to the results, PROM outperforms Q-routing by a substantial margin, achieving the highest efficiency under bursty traffic with a PM of 38.42%. DyAD performs best for CBR traffic, with a PM of 34.97%, compared to XY's PM of 33.64%. The results show that traffic conditions affect the algorithm's applicability. DyAD performs well under constant loads and PROM is well suited for unpredictable traffic.

Keywords: Network on Chip, Mesh Topology, XY Routing, PROM Routing, Q Routing, Dyad Routing, Throughput, Latency.

1. Introduction

The growing demand to integrate technologies like IoT, robotics, computing, communication, and video processing into a single chip has made System on Chip (SoC) a central element in modern electronics. SoC has been used in many applications ranging from smartphones to spacecraft. As multi-core advancements allow more memory and processing units per chip, the complexity of on-chip interconnects has become a critical bottleneck. To address this issue, Network on Chip (NoC) provides a scalable and reliable communication backbone for SoCs, enabling high performance in chip multiprocessors and complex systems through its modular and efficient design.

NoC works as a communication or transport system for the data exchange between various modules, like specialized IP cores, DSPs, memory, and general processors. In this architecture, each tile includes a processing core, a network interface, and a router. Topology is the arrangement of nodes and links on the chip, directly affecting network latency and bandwidth. Common NoC topologies include mesh, ring, butterfly, torus, and star, with the 2D mesh being the most widely used for its simplicity and scalability [1, 2]. Cores are interconnected through a mesh of routers and communicate via packet-switched techniques. These routers are connected through point-to-point links, allowing efficient data transfer across the chip [3]. Each router independently determines the routing path, enabling data packets to travel from source to destination through multiple routes. This routing strategy enhances parallelism and maximizes the performance potential of the NoC [4-6]. The overall NoC performance depends on topology, routing strategy, and switching technique [7].

A network's performance in NoC is determined by latency, throughput, and power dissipation. These parameters are influenced differently by the chosen routing technique employed in a NoC architecture. Selecting a routing algorithm is crucial, as no single routing technique offers optimal performance across a wide range of applications [8, 9]. Since traffic patterns vary by applications, a routing algorithm that excels in throughput may result in an increase in latency or power consumption in another. These algorithms aim to avoid congestion, deadlock, and starvation while maximizing throughput and minimizing latency. Optimizing one metric may often compromise the performance of other parameters. Additionally, the same routing strategy may produce inconsistent results across different traffic [1, 2, 10, 11, 12]. As a result, a universal routing solution for all applications is impractical. Instead, routing decisions can be made to achieve specific goals, such as minimizing path length (minimal vs. non-minimal) or selecting the right data transmission path (source vs distributed routing) [3, 13]. The goal of this study is to contribute to the advancement of routing for communication in NoC. The major contributions of this research work are as follows

- The core contributions of the proposed work lie in the evaluation of four routing algorithms, namely XY, Q, PROM, and DyAD for CBR and bursty traffic on a 9 x 9 mesh topology under different load injections.
- The evaluation focuses on key network parameters such as throughput, latency, and power consumption and determination of most optimal routing for a CBR and bursty traffic type.

This research work are presented in the following sequence. The literature review is covered in Section 2, several routing algorithms are covered in Section 3, the experimental setup is detailed in Section 4, simulation results are explained in Section 5, and the work is concluded in Section 6.

2. Literature Review

The characteristics of a routing algorithm is illustrated in figure 1. An effective routing algorithm in NoC systems should ensure strong connectivity, flexibility, load balancing, deadlock and livelock freedom, fault tolerance, starvation avoidance, high performance, and low latency.

2.1 Routing Procedure and Classification of Routing Algorithms

In Figure 2, Initially, the routing tables are set up, and virtual channels are allocated for each network link to enable handling of multiple concurrent packets. The processing elements (PEs) then begin packet generation, with each packet containing a destination address. Once a packet is generated, the source PE refers to its routing table to identify the next hop and selects an available virtual channel for transmission. If all channels are busy, the packet waits in a buffer. The selected route is followed through intermediate routers until the packet reaches its destination. Throughout the process, mechanisms for deadlock prevention, congestion control, and error handling ensure reliable delivery. Periodic updates to routing tables and adaptive routing strategies help optimize performance based on network traffic patterns, ensuring efficient communication across the NoC.

The classifications of routing algorithms are based on several key criteria, such as decision-making, adaptivity, path optimality, and implementation mechanism, as shown in Figure 3 [14]. For more details of different routing algorithms refer table 1.

2.1.1 Based On Communication Mode

Routing techniques are classified into unicast or multicast. Unicast routes packets to a single destination, while multicast routes the same packet to several destinations [4]. In NoC, unicast routing is a practical choice due to the presence of direct communication links between different components within the chip. For 3D Mesh, the Minimal and Adaptive Routing (MAR) algorithm distributes the unicast and multicast traffic more efficiently over the network [15].

2.1.2 Based on Routing Decisions

Routing algorithms can be broadly categorized as deterministic and adaptive routing algorithms [4]. Deterministic routing involves the forwarding of packets along predetermined routes. Therefore, during network congestion, packets are unable to use other routes, resulting in a decline in performance [3]. XY routing is an example of deterministic routing. While deterministic routing algorithms are simple to implement, they fail to evenly distribute the load across the network in scenarios involving non-uniform or bursty traffic sources [4]. Due to the fact that packets always arrive at their destination in the correct order, they are an excellent choice for real-time systems [5]. Adaptive routing algorithms dynamically select routing paths based on real-time network conditions, including traffic load, link capacity, congestion, and faults in the network. Adaptive routing offers better performance than deterministic methods under unpredictable traffic patterns. Adaptive routing is used in congested networks with nonuniform traffic because it may disperse traffic around the network to alleviate congestion [16]. However, it also introduces architectural complexity and implementation overhead. Therefore, these algorithms incur more costs, occupy more space, and use more power. Its operation involves dynamic arbitration and next-hop selection [7]. Notable adaptive routing techniques include DyAD [17], Q-routing [6], Dimension-Ordered Routing with Adaptive Turn (DTAR), Odd-Even Turn Model (OETM) [18], and Minimal Adaptive Routing (MAR) [15]. The key advantage of adaptive routing lies in its ability to alleviate congestion by efficiently utilizing multiple available communication paths.

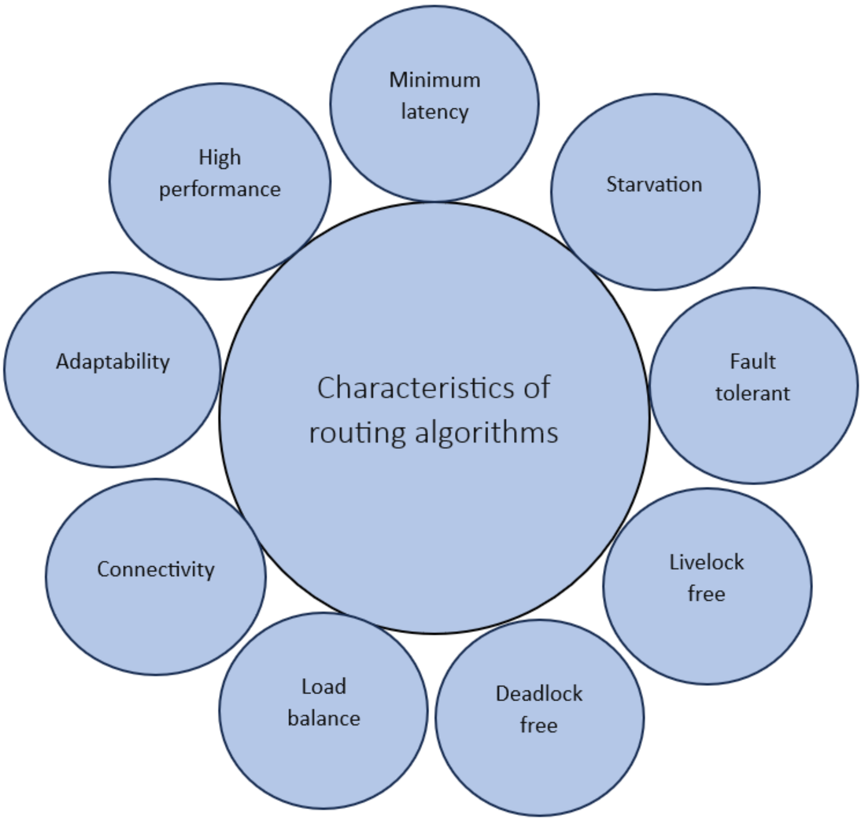

Figure 1. Characteristics of routing algorithms

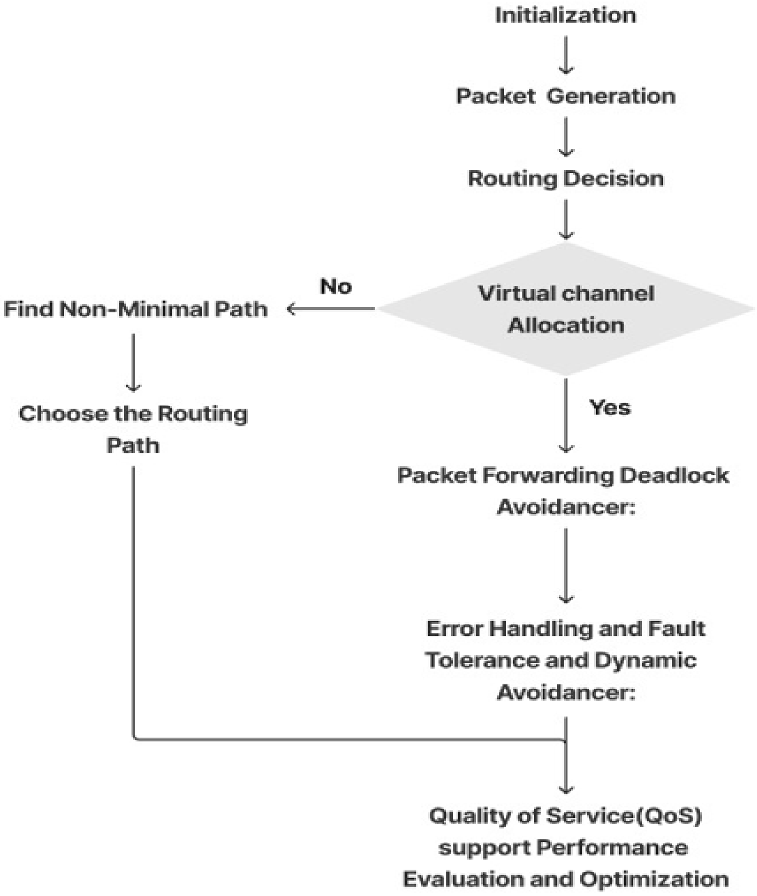

Figure 2. Routing Procedure

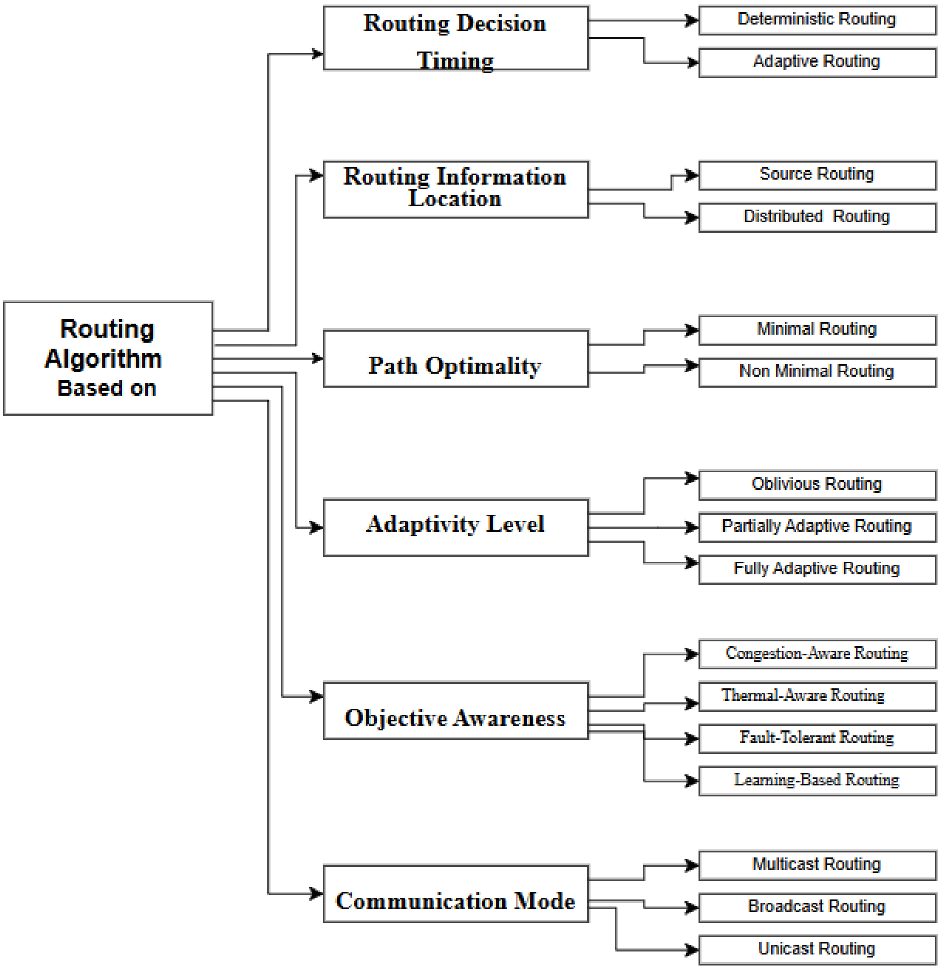

Figure 3. Classification of Routing Algorithms

2.1.3 Based on Routing Information Location

Routing in NoC can be classified as source routing and distributed routing based on routing information location. In source routing paths are pre-computed and stored in the packet header, but this adds overhead and lacks path adaptivity in the event of fault and congestion.

In distributed routing the destination address is embedded in a packet's header, and the route is determined dynamically by the intermediate routers, making it more fault tolerant but also making it more complex in terms of logic and memory.

2.1.4 Based on the adaptability level

The routing is classified as oblivious [19], fully adaptive and Partially adaptive. Oblivious routing, ignores network condition while routing packet eg. O1turn. Fully adaptive techniques, selects congestion free path regardless of the distance eg. DyAD and turn models. Partially adaptive routing algorithms restrict packets from using all of the shortest routes [20, 8] while balancing benefits of fully adaptivity and deterministics routing eg. West– First, Negative-First.

2.1.5 Based on Path optimality

On the basis of the path optimality two types of routing are Minimal and Non-Minimal. Minimal routing methods simply use the shortest path but risks deadlock [20] eg. XY, Odd-Even. Non-minimal routing also allows longer path, offering load balancing and fault tolerance with the risk of livelock eg. OCRA (Oblivious Congested Region Avoiding Routing Algorithm) [21], Valiant's Randomized Routing. Partially adaptive and completely adaptive routing algorithms may be differentiated based on their minimization or non-minimization characteristics [20, 8].

2.1.6 Based On Objective Awareness

NoC routing can be congestion aware, thermal aware, fault tolerant, learning based, and designed to handle deadlock or livelock. Each routing technique addresses one or some of the problems. In congestion-aware methods, adjust paths by using congestion information to improve buffering, packet allocation, and performance. eg. DyXY [22], Q-RASP [23], CAAR [24], AFTC [25]. In thermal awareness routing, e.g., TTQR, reroutes traffic to avoid hotspots, using Q-tables to balance thermal load and temperature across 3D NoC [15].

Backtracking is an example of a fault-tolerant routing technique that ensures reliable packet delivery around faulty nodes or links, or by selecting alternative paths to reach the destination [9]. Distributed fault-tolerant routing protocols include XY-YX and West-First Fault-Tolerant routing. Advanced approaches like NoCAlert- supported algorithm, machine learning Methods (Q-Learning algorithm, DRL routing) adaptively make routing decisions [11].

Deadlock free routing [12] prevents cyclic dependencies eg. XY, Turn models. Three versions of the turn model are west-first, north-last, negative-first. While recovery methods, resolves them by using timeout or escape path eg. Progressive adaptive routing. Approaches like lucrative algorithms favours shortest progress while misrouting algorithms temporarily divert packets or flits away from the destination for load balancing or fault avoidance. XY-YX avoids deadlocks and livelock in NoC. The deadlock avoidance methods minimize path restrictions to improve performance without added hardware complexity [26].

Implementation wise, routing algorithms may be classified as lookup tables or Finite State Machine (FSM) [27]. Lookup tables are commonly used in software implementations as storage of a lookup table is possible at each node and can be easily modified. Finite State Machines (FSMs) may be implemented in either software or hardware. These approaches can be further categorized as Adaptive Routing and Deterministic Algorithms based on the flexibility [28]. Application Specific Routing is customized routing algorithms for a particular use case NoC based SoC, optimizing communication patterns between several cores [13].

Hybrid routing combines different routing methods to balance between several aspects like latency, throughput, power consumption, and fault tolerance. PcaDAHR is an example of a hybrid routing that combines deterministic and adaptive routing [12]. PAAD (Partially adaptive and deterministic routing) switches between partially adaptive and deterministic routing based on network congestion [29].

3. NoC Routing Algorithms - XY, PROM, Q and DyAD

Numerous widely utilized routing algorithms exist for NoC topologies. This section explains about routing algorithms used in this proposed work.

3.1 Dimension Order Routing (Dor) or XY Routing

Packets are first directed along the X-axis and then the Y-axis via Deterministic Routing (DOR), which is often implemented as XY routing. In mesh/torus NoCs, it is straightforward and effective, although it is unable to manage hotspots and congestion [30, 31]. In this routing, a router with coordinates (Current_x, Current_y) forwards packets first along the X-dimension (East if smaller, West if larger) until Current_x is equal to Destination_x then along the Y-dimension (South if smaller, North if larger) until it reaches (Destination_x, Destination_y) [32].

3.2 Q-Routing

Boyan *et al*. [5] describe a Q-routing method that includes a reinforcement learning module in each router of network-on-chips (NoCs). Reinforcement routing is a machine learning-based routing algorithm. In RL, an agent interacts with its surroundings through actions, states, and rewards. Reinforcement learning (RL) makes decisions all the time and gets information from its surroundings, rather than waiting for clear feedback on whether its actions are right or useful [33]. Each intermediate router acts as an agent, using Q values to predict and update cost through trial and error. This enables dynamic fault tolerant routing that adapts to real time congestion. Q-learning [17] and Deep Q-Network (DQN) [27] are common methods for RL. Q-routing provides a way to collect global information about congestion from neighboring nodes and project cost back to the current node, this helps to improve the performance.

In Q-Learning, the goal is for an agent to learn a function (s, a), where s is the current state, a is the action, and (s, a) is the total prize that action a won in state s. The change in the N-value depends on applying the Bellman equation over and over again. This equation shows how the Q-value of the current state is related to that of the next state. where α denotes the learning rate, γ represents the discount rate, and $maxQ(s', a')$ signifies the maximum Q-value for the subsequent state as shown in equation 1. The Q-table is used in Q-Learning to retain Q values.

$$(s, a) \leftarrow (s, a) + \alpha(r + \gamma maxQ(s', a') - Q(s, a)) \qquad (1)$$

Q-routing outperforms traditional methods by learning from experience and maintains Q-values that estimate congestion in the network and update through cost function and estimates from the neighbor nodes. The Q value is constantly updated. Its main drawback is slow computation due to the large size of Q-tables. This drawback is overcome by C-routing [34] and Bi-LCQ [17] to reduce the Q-table size. In Q-routing, the Epsilon (ε)-Greedy approach is utilized to balance exploration and exploitation. Where ε with a certain probability explore new routes and exploit already known good routes. A fixed (epsilon) might not be suitable for diverse network loads or topologies, as exploring too much can worsen congestion, while exploring too little can trap routing policies in suboptimal paths.

The learning rate α determines how strongly new rewards update Q-values. Proper tuning of ε and α allows the policy to adapt to network conditions and eventually converge to optimal routing decisions. The value of alpha ranges from 0.1 to 10. If alpha value is zero means no learning is performed. Higher the value of alpha means more learning is performed [18]. For example, an agent policy may maximize the cumulative reward by taking action a^* = max Q(s_t,a) at each step. In most implementations, an ε-greedy policy is used where an agent can query the table to either select the greedy action a^* to optimize return or randomly select an action with a probability ε to explore the environment for learning. In this way, the ε-greedy policy can explore alternative solutions by selecting nonoptimal actions with some probability. Given sufficient update coverage of all state-action pairs, the policy eventually converges to the optimal state-action policy.

3.3 Path-Based, Randomized, Oblivious, Minimal Routing algorithm (PROM)

While selecting the routes, deterministic and oblivious routing both will ignore network congestion. Techniques such as flooding and turn models (West-First, North-Last, Negative-First) are examples of oblivious routing. Both may result in longer delays in congested areas since they only consider source to destination routes and ignore traffic conditions. For lower latency, adaptive routing is therefore necessary to dynamically modify routes according to traffic load [35, 22].

Table 1. Comparison of different routing algorithm

Algorithm	**Path definition**	**Path length**	**Based on**	**Topology**	**ref**
XY	Deterministic	Minimal	-	2D Mesh	Paper [38]
Turn Model	Partially adaptive	Minimal & Non-minimal	-	2D Mesh	Paper[1]
West-First	Fully adaptive Partially adaptive	Minimal & Non-minimal	Turn Model	2D Mesh	Paper[2]
North-Last	Partially adaptive	-	Turn Model	2D Mesh	Paper[10]
Negative -first	Adaptive	Minimal	Turn Model	2D Mesh	Paper[11]
XY-YX	Deterministic	Minimal	XY	2D Mesh	Paper[12]
DyAD	Deterministic & Adaptive	Non-Minimal	OE	2D Mesh	Paper [39]
Q- Routing	Adaptive	Non-Minimal	-	2D Mesh	Paper [40]
PROM	Partially adaptive	Minimal	-	2D Mesh	Paper[41]
Thermal aware Dy-XYZ	Fully adaptive	Minimal	XYZ	3D Mesh	Paper[42]
RL Lookahead	Adaptive	Minimal	RL	3D Mesh	Paper[25]
DATRA (Dynamic Adaptive Threshold Routing Algorithm)	Adaptive	Minimal & Non-Minimal	universal global adaptive load-balanced routing (UGAL)	2D Mesh	Paper[41]
TTQR	Adaptive	Minimal	Q-routing	3D Mesh	Paper[26]
RLARA	Adaptive	Minimal & Non-Minimal	Q-routing	3D mesh	paper[25]
PAAD	Adaptive	Minimal & Non-Minimal	Hybrid(Turn model and DOR)	2D Mesh	Paper[33]
OCRA	Adaptive	Non-Minimal	Static configuration of routers	3D mesh	Paper[11]
PcaDAHR	Adaptive	Minimal	Hybrid (deterministic and adaptive)	2D Mesh	Paper[32]

A series of minimal, path-diverse, oblivious routing algorithms for NoCs is called Path-based, Randomized, Oblivious, Minimal (PROM) [36]. By adopting randomization, where each hop is selected locally by fair arbitration without knowledge of the global context, it balances load. PROM supports variations including O1 TURN PROM, Uniform PROM, Parameterized PROM, and PROMV and requires a minimum of two virtual channels to prevent deadlocks [31, 32, 28].

3.4 Dynamic Adaptive Deterministic (DyAD)

The Dynamic Adaptive Deterministic (DyAD) is a routing methodology that combines the advantages of both deterministic and adaptive routing approaches. Based on the congestion in the current network DyAD switches between adaptive and deterministic modes [37]. To toggle between adaptive and deterministic, each router in the DyAD continuously monitors its local network. Each router in DyAD is assigned a congestion flag to serve as an indicator of network congestion. This flag is sent to all interconnected routers. On receiving the flag, the mode controller activates the adaptive mode of the router.

The minimum route algorithm of odd-even was used to prevent network livelock and reduce energy usage. The integration of adaptive and deterministic routing techniques serves to prevent both livelock and deadlock. The primary benefit of using deterministic routing is in its simplicity of the router design structure. Deterministic mode is used when the network is not congested; this leads to low latency. The DyAD router switches to adaptive mode when the network is congested to avoid congested routing paths; which leads to higher throughput in the network [17].

3.5 Comparison Analysis of Routing Algorithms Based on Adaptivity

Due to the fact that route methods can change, Table 1 shows the comparison of several routing algorithms used in NoC.

4. Experimental Setup

Alongside the aforementioned routing methodologies, the design of routing algorithms in NoCs has been governed by the ideas of avoiding congestion, deadlock, and starvation, while maximizing throughput and minimizing latency. This study presents a traffic-driven routing algorithm selection approach that adjusts to network performance needs including low latency, high throughput, and low power consumption.

Nirgam simulator, also known as NoC Interconnect Routing and Application Modeling, is a systemC based discrete event cycle model that uses Orion to perform power calculations [43]. This simulator supports mesh and Torus topology. Nirgam simulator uses the Orion analytical model by calling its API inside the router and power calculation model. In Orion, total link power is the sum of leakage power and dynamic power. For router power consumption buffer, crossbar, arbiter and leakage power is considered. Router power and link power are wrapped in the Orion power model. The different parameters used in the Orion model for simulation are Tech node (Process node) is 90 nm, Supply voltage (VDD) is 1.2 and frequency is 1GHz. Nirgam simulator gathers statistics of number of flits, buffer read/ write, link traversals. These statistics are passed on to the Orion model. Orion generates power consumption. Watt is used to measure power.

Simulation is performed on 9 x 9 mesh topology using a wormhole switching mechanism. Simulation runs for 1000 cycles with a warm up of 5 cycles and a target for 300 cycles [18]. Latency parameter is evaluated per flit basis, per packet basis and overall latency. For analysis, the routing algorithms XY, PROM, Q and DyAD are applied on mesh topology in the Nirgam simulator. Performance is checked for each channel separately in the Nirgam simulator. Injection rate is varied from 10 to 100 %. If the injection ratio is 30% means only 30 % of the maximum bandwidth is utilized. Throughput is measured in Gbps in the X and Y direction separately. CBR and Bursty traffic is considered for the analysis. Alpha is the learning rate for Q-routing. Alpha is taken as 0.8, which is constant [44]. Table 2 shows the simulation parameters used for this analysis work.

5. Results and Discussion

The injection rate is referred to as the number of flits that are injected into the network during each clock cycle. Latency is the time elapsed for data to reach the destination from its source. Average message latency depends on the injection rate. As the injection load increases contention and latency will increase. Latency increases exponentially at the saturation point of injected load. Latency is affected by traffic and congestion. Throughput informs how much data a network can send and receive. Bits per second of good packet deliveries are used to measure throughput. The most important thing in NoC design is power. There are two types of power namely static power and dynamic power. Link power is dynamic and dissipated by the links in the topology. Power depends on the number of links a packet visits from its source to its destination. Less power is consumed if a packet traverses less links. Power consumed by the buffers and other components inside the router even when they are not active is static leakage power.

Metrics used to assess the efficiency of routing algorithms include optimal route for packet transmission, average delay, and throughput. Average throughput divided by average latency is called Performance Metrics (PM) as indicated in equation 2. Performance Metric helps to compare routing algorithms more effectively. PM increases when throughput increases and latency decreases. A High PM value signifies that more data is being delivered efficiently in minimal delay. It reflects high effectiveness and efficiency of the routing algorithm. This is more useful for adaptive routing like DyAD. PM is unitless. The factor 100 is used for normalization or easier comparison. For each channel PM is calculated by equation 2.

Performance Metrics (PM) = (average throughput * 100) / (average latency) (2)

5.1 CBR Traffic

Figure 4 to 7 are the Nirgam simulator generated results. For bursty traffic, the power consumed by the routers in the X and Y direction at 100% load is clearly indicated in figure 4. The blue colour indicates less power is consumed by most of the routers in DyAD routing. The throughput for bursty traffic at 40% load with PROM routing is shown in figure 5 and the throughput of CBR traffic at 100% load with DyAD routing is depicted in figure 6. Nirgam simulator uses XY plane to represent routers in the X and Y direction and vertical axis to represent power, latencc and throughput respectively. Nirgam simulator shows throughput channel wise in the X and Y direction. In figure 5 and 6, the left side plot shows the throughput of the routers in the Y direction which is depicted by green colour. Similarly in figure 5 and 6, the right side plot shows the throughput of the routers in the X direction which is depicted by red colour. The maximum throughput is achieved at 40% load for bursty traffic by PROM routing and for CBR traffic it is achieved by DyAD routing at 100% load. Figure 7 shows the power consumption of CBR traffic at 50% load with DyAD routing.

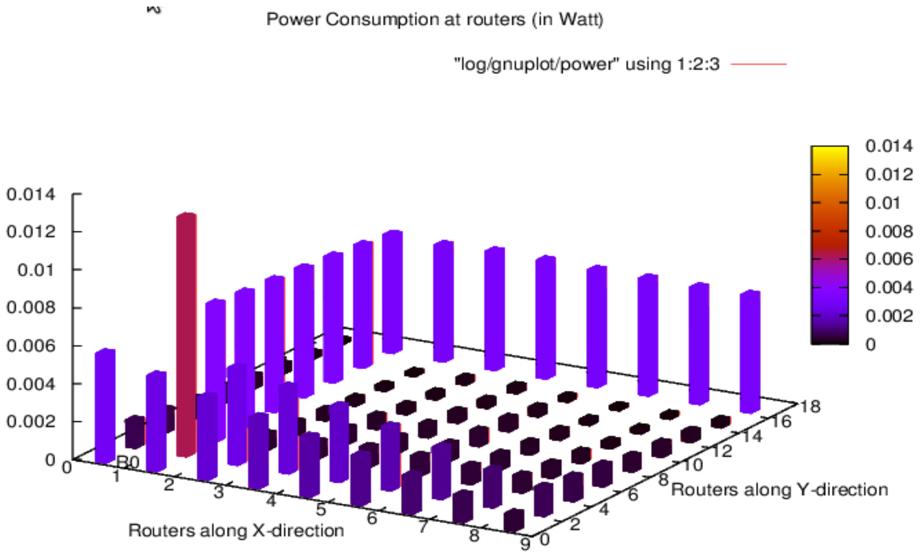

Figure 4. Power consumption of bursty traffic at 100% load with DyAD routing

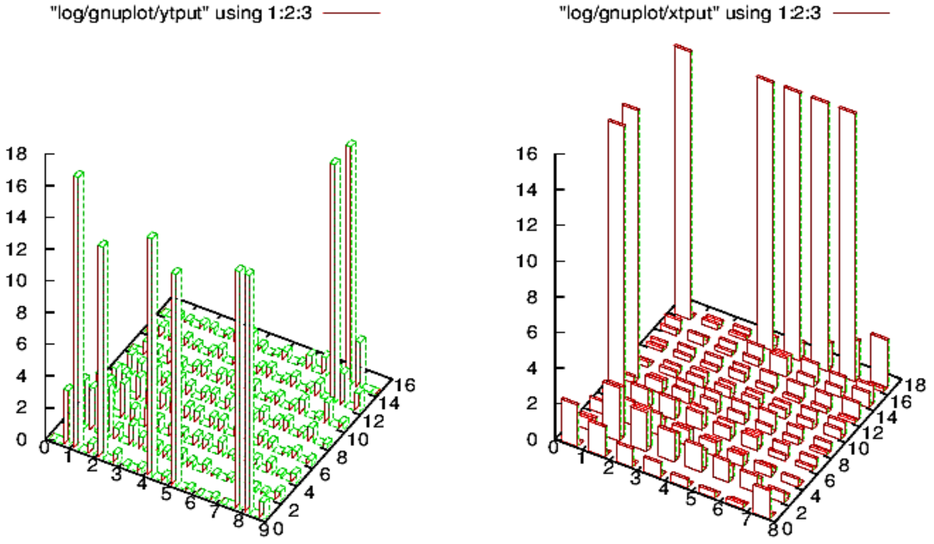

Figure 5. Throughput for bursty traffic at 40% load with PROM routing

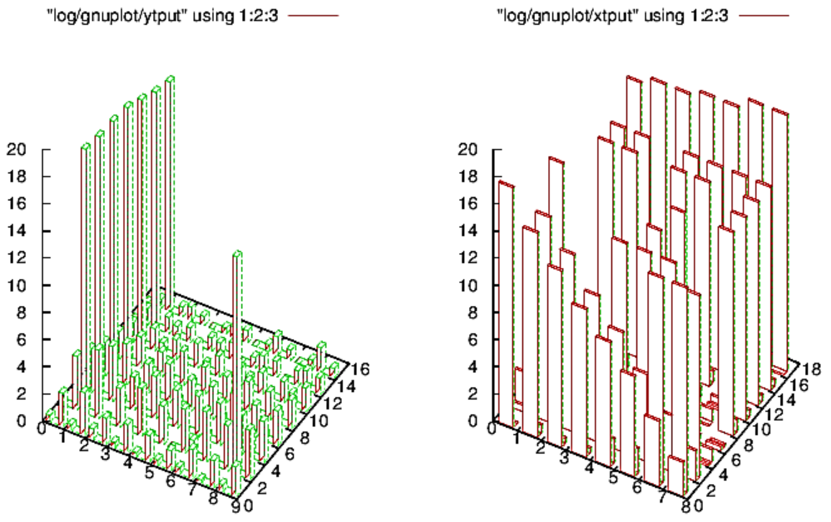

Figure.6. Throughput of CBR traffic at 100% load with DyAD routing

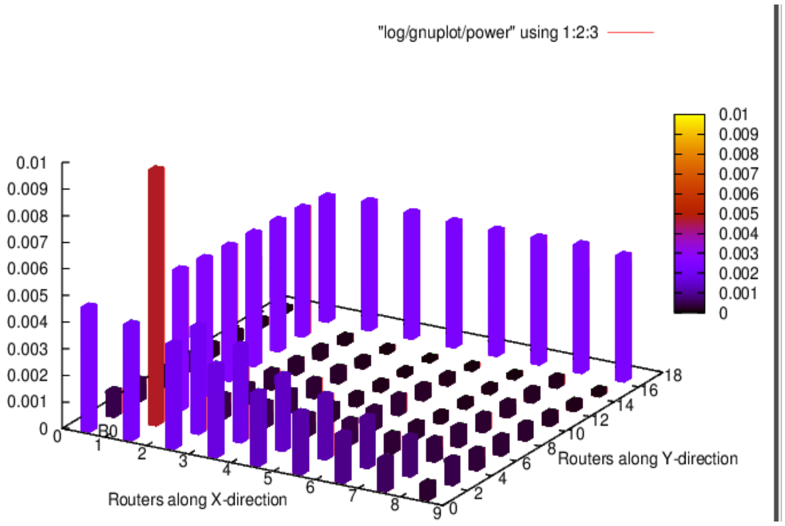

Figure 7. Power consumption of CBR traffic at 50% load with DyAD routing

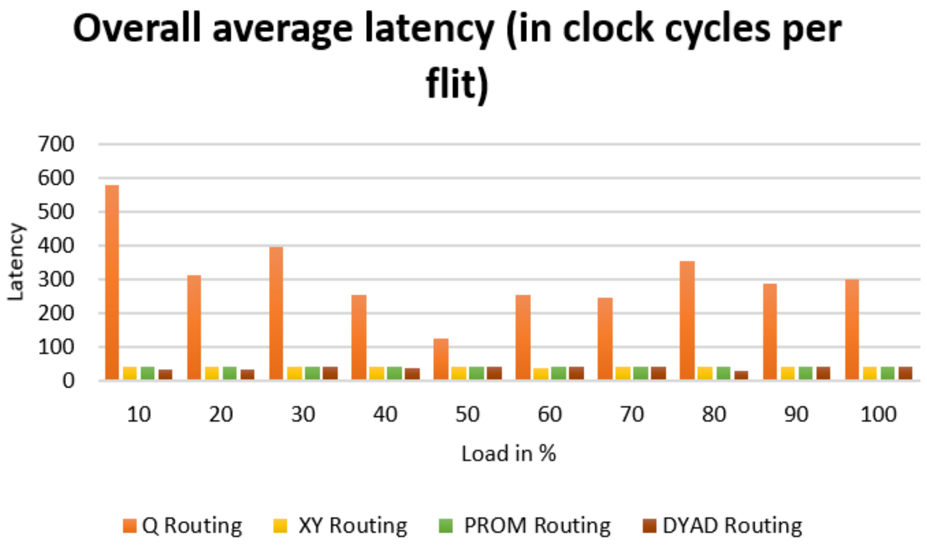

Figure 8. Overall average latency (in clock cycles per flit) of Q, XY, PROM and DyAD routing for CBR traffic

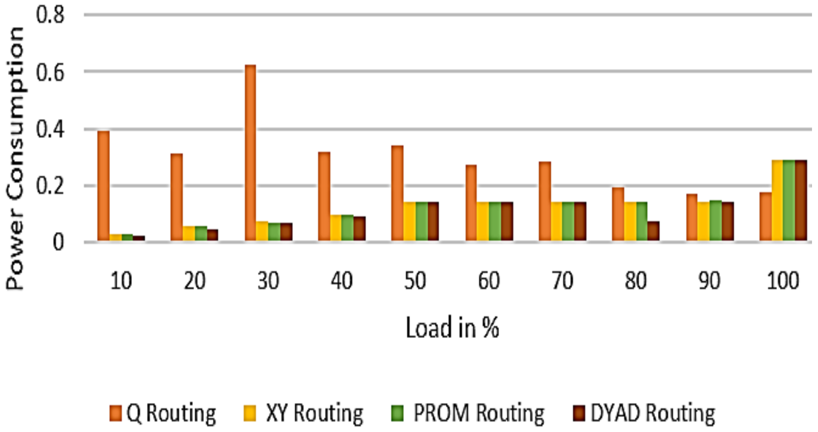

Figure 9. Power consumption (per flit) of Q, XY, PROM and DyAD routing for CBR traffic

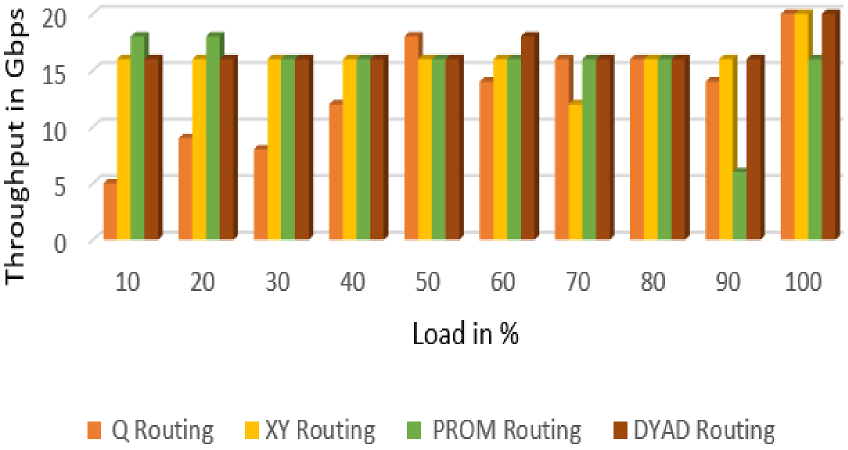

Figure 10. Y direction throughput (average number of packets delivered per cycle per node) of CBR traffic with Q, XY, PROM and DyAD routing

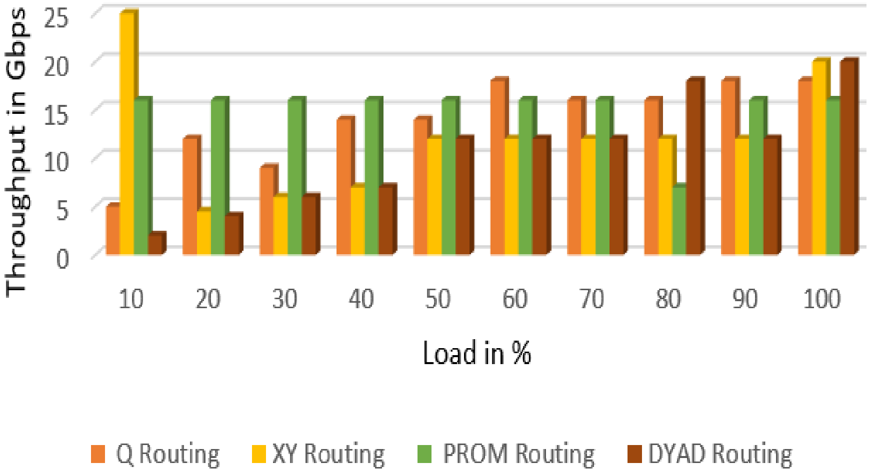

Figure 11. X direction throughput (average number of packets delivered per cycle per node) of CBR traffic with Q, XY, PROM and DyAD routing

Figure 8, 9, 10, and 11 shows simulation output for 9 x 9 mesh topology with Q routing, XY routing, PROM and DyAD routing for CBR traffic. Figure 12, 13, 14 and 15 shows simulation output for 9 x 9 mesh topology with Q routing, XY routing, PROM and DyAD routing for bursty traffic. From figure 8 and figure 12 it is clear that overall average latency increases as the payload increases.

In case of CBR traffic from figure 8 it is observed that overall average latency is more for Q routing as compared to other routings as Q routing is adaptive so more complex as compared to other routing[a]. Overall, DyAD route has the least amount of delay as it follows deterministic for non congested route and adaptive routing for congested route. Additionally, avoiding congested paths helps in reducing the delay. The average delay for XY and PROM is almost the same.

The overall power consumption increases as more load is injected into the XY, PROM, and DyAD routes shown in figure 9 and figure 13. Since power consumption depends on the number of links visited. For Q routing initially power consumption is more and later on it decreases as Q routing is reinforcement routing which initially learns all paths and then exploits them [26]. The Q routing consumes more average power as compared to other routing techniques as Q-routing is more complex. The average power consumption by the DyAD routing in case of CBR traffic is less as compared to other routing techniques. In case of DyAD as it uses a deterministic routing approach when low congestion, which helps in reducing router activity and buffer usage as minimal path is used. Since a minimal path is used, latency is reduced. Similarly when the network is congested then DyAD switches to an adaptive approach which helps in preventing packet delay and retransmission, so efficient use of links and buffers helps to achieve reduction in power consumption as compared to Q- routing which is adaptive in nature [43] as shown in figure 9 . whereas XY and PROM consume almost the same amount of the average power.

In case of CBR traffic as observed from the figure 10 the average throughput in the Y direction for DyAD is the best as latency is reduced so more throughput can be achieved. The XY routing has better average throughput than Q and PROM routing but it is less than DyAD routing.

From figure 11, the throughput of PROM in X direction routing is more than other routings. The Q routing has better average throughput than XY and DyAD routing but it is less than routing PROM. The balancing of deterministic and adaptive routing in DyAD outperforms other routing techniques in case of CBR traffic [40]. DyAD offers more balance between average latency, throughput and power consumption as compared to deterministic XY, adaptive Q routing and PROM routing. The threshold in DyAD routing reflects the balance between deterministic and adaptive routing. A lower threshold implies a preference for deterministic routing, while a higher threshold favors adaptive decisions based on network conditions.

5.2 Bursty Traffic

For Bursty traffic from the figure 12 it is observed that overall average latency for DyAD routing is less compared to the other routing techniques as explained in section 5.1.

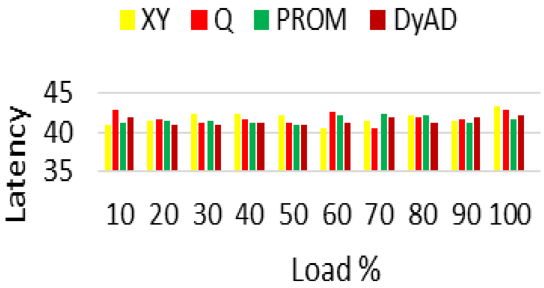

Figure 12. Overall average latency (in clock cycles per flit) for Bursty traffic with XY, Q, PROM and DyAD routing

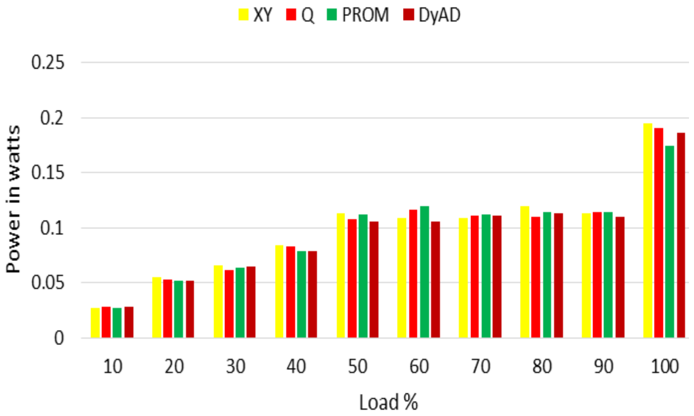

Figure13. Power consumption (per flit) of XY, Q, PROM and DyAD routing for bursty traffic

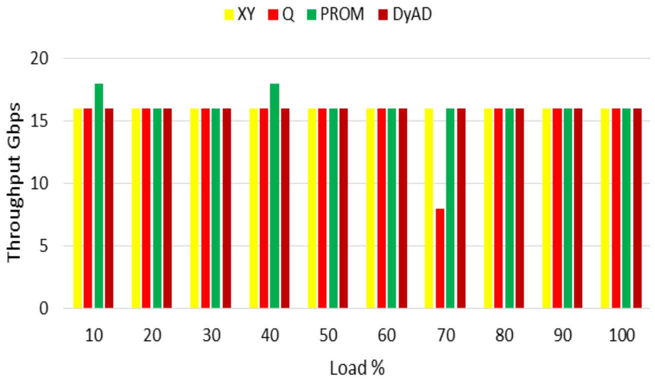

Figure14. Y direction throughput (average number of packets delivered per cycle per node) of bursty traffic with XY, Q, PROM and DyAD routing

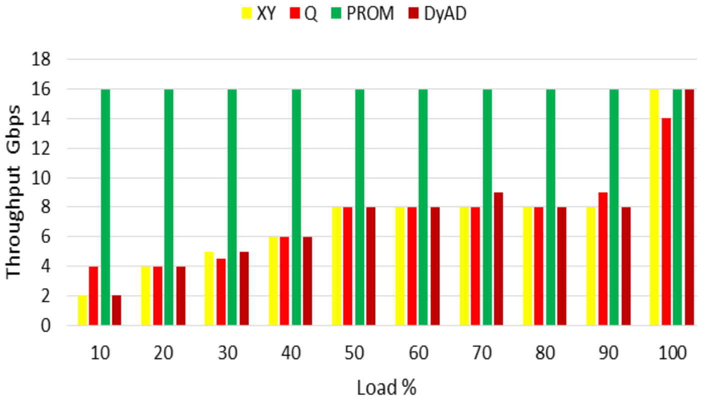

Figure15. X direction throughput (average number of packets delivered per cycle per node) of bursty traffic with XY, Q, PROM and DyAD routing.

Table 3. Comparison for PM for CBR and bursty traffic for Q, XY, PROM and DyAD routings.

Routing methods	CBR Traffic PM	Bursty Traffic PM
Q	4.35%	17.97%
XY	33.64%	18.09%
PROM	36.74%	38.42%
DyAD	34.97%	18.51%

Again it is observed from the figure 13 as the injection ratio increases, the power consumption for the bursty traffic also rises. From figure 13, Less power is consumed by DyAD routing for bursty traffic as compared to other routing techniques as explained in section 5.1.

For bursty traffic, from figure 14 and figure 15 it is observed that PROM routing offers more average throughput in both X and Y direction as compared to other routing methods. PROM predefines multiple minimal paths and randomly selects among them, ensuring minimal routing. With path diversity of minimal paths, traffic is distributed reducing the congestion in the network. The minimal path and randomization helps to increase the throughput [12]. The Q, XY and DyAD routing techniques are offering more average throughput in the Y direction as compared to the X direction as observed from figure 14 and figure 15.

Table 3 shows the comparison of PM for CBR and bursty traffic for Q, XY, PROM and DyAD routings. The PM of Q routing for CBR traffic is 4.35% which is less as compared to other routing methods in case of CBR and Bursty traffic as observed from table 3. The PM of XY routing is 33.64% for CBR traffic which is more than 18.09% for bursty traffic. The PM of bursty traffic for PROM routing is 38.42% which is more than the PM for CBR traffic which is 36.74%. The PM of DyAD routing for CBR traffic is 34.97% which is more than 18.51% of bursty traffic. The XY and DyAD routing out performs for CBR traffic as compared to bursty traffic. For Bursty traffic Q routing and PROM performs better than other routing techniques. Deterministic routing performs better in case CBR traffic whereas adaptive routing performs better for bursty traffic.

6. Conclusion & Future Scope

The latency, power consumption, and throughput of PROM routing surpass those of other routing methods. The PROM routing does not consider real time traffic. So under hotspot traffic it cannot reroute packets to avoid congested paths. The packets pile up near the hotspot node causing delay. Due to the delay at hotspot overall throughput will reduce. XY is a deterministic routing technique that is independent of the current network condition. As a result, the route used by each source-destination pair remains constant; it cannot be altered at the time of congestion. Although an oblivious routing method is designed to provide communication without deadlock or livelock, with improved throughput and reduced latency. The Q routing technique is a congestion aware routing algorithm that achieves improved throughput, reduced power consumption but as the size of the network grows the size of the Q-table increases significantly. DyAD has a limited adaptivity so it is better than deterministic but still falls short of fully dynamic algorithms like Q-routing. DyAD has a more complex design as it has to decide the routing nodes.

Network-on-Chip (NoC) routing algorithm selection depends on multiple factors including network architecture, traffic patterns, and performance criteria, with no single universal solution optimal for all applications. While deterministic methods like XY routing provide consistent paths, techniques like PROM routing offer superior performance and dynamic approach of Q-routing provide congestion-aware capabilities but may increase packet delay. The study demonstrates through simulation experiments that application-specific routing decisions are essential, advocating for traffic-driven dynamic routing systems that can adapt to varying network requirements. The proposed approach focuses on avoiding congestion, deadlock, and starvation while optimizing latency, throughput, and power consumption, concluding that NoC architectures require flexible, application-driven dynamic routing systems rather than fixed routing strategies to achieve optimal performance across different scenarios. Future research should concentrate on creating multi-objective optimization frameworks that concurrently balance latency, power consumption, and throughput for a variety of applications, as well as machine learning-based adaptive routing algorithms that can forecast the best routes in real-time.

References

[1] L. Benini, D. Bertozzi, Network-on-chip architectures and design methods, IEE Proceedings-Computers and Digital Techniques, 152(2), (2005) 261-272. https://doi.org/10.1049/ip-cdt:20045100

[2] A. Gogoi, B. Ghoshal, A. Sachan, R. Kumar, K. Manna, Application driven routing for mesh based Network-on-Chip architectures. Integration, 84, (2022) 26-36. https://doi.org/10.1016/j.vlsi.2021.12.008

[3] A.S.K. Sandhiya, S. Saranya, U. Harsha, Forecasting website traffic using prophet time series model, International Research Journal of Multidisciplinary Technovation, 1(1), (2019) 56-63. https://doi.org/10.34256/irjmt1917

[4] K. Ahmad, M.A.J. Sethi, Review of Network on Chip Routing Algorithms, EAI Endorsed Transactions on Context-Aware Systems and Applications, 7(22), (2020) e5. https://doi.org/10.4108/eai.23-12-2020.167793

[5] W. Zhang, L. Hou, J. Wang, S. Geng, W. Wu, (2009) Comparison Research between XY and Odd-Even Routing Algorithm of a 2-Dimension 3X3 Mesh Topology Network-on-Chip, 2009 WRI Global Congress on Intelligent Systems, IEEE, Xiamen, China. https://doi.org/10.1109/GCIS.2009.110

[6] T. Deeka, (2021) Adaptive dynamic packet routing on internet networks based on reinforcement learning approach, University of Southampton, Master's Thesis, 119. https://eprints.soton.ac.uk/453034/

[7] S. Choudhary, S. Qureshi, Performance evaluation of mesh-based NoCs: Implementation of a new architecture and routing algorithm, International Journal of Automation and Computing, 9(4), (2012) 403-413. https://doi.org/10.1007/s11633-012-0661-1

[8] K.K. Paliwal, J.S. George, N. Rameshan, V. Laxmi, M.S. Gaur, V. Janyani, R. Narasimhan, Implementation of QoS Aware Q-Routing Algorithm for Network-on-Chip. In Communications in computer and information science, (2009) 370-380. https://doi.org/10.1007/978-3-642-03547-0_35

[9] M. Ahmed, R. Kumar, (2012) Parameterized path-based, randomized, oblivious, minimal routing in 3D mesh NoC. TENCON 2012 IEEE Region 10 Conference, IEEE, Cebu, Philippines. https://doi.org/10.1109/TENCON.2012.6412341

[10] C. Wang, W.H. Hu, S.E. Lee, N. Bagherzadeh, Area and power-efficient innovative congestion-aware Network-on-Chip architecture. Journal of Systems Architecture, 57(1), (2011) 24-38. https://doi.org/10.1016/j.sysarc.2010.10.009

[11] T.A. Kumar, R. Rajesh, P. Sivanainthaperumal, Performance Analysis of Noc Routing Algorithms For 5 × 5 Mesh Based Soc. ICTACT Journal on Microelectronics, 1(4), (2016) 141-146. https://doi.org/10.21917/ijme.2016.0028

[12] B.K. Soni, G. Parmar, Analysis of Different Routing Algorithm for 2D-Torus Topology NoC Architecture under Load Variation. International Journal of Advanced Engineering Research and Science 3(12), (2016) 250-255. https://doi.org/10.22161/ijaers/3.12.44

[13] M.S. El Sayed, S.A. Salem, M.H. Awadalla, A.M. Mostafa, A power efficient, oblivious, path-diverse, minimal routing for mesh-based networks-on-chip. International Journal of Computer Science Issues, 9(2), (2019) 339.

[14] A. Javed, J. Harkin, L. McDaid, J. Liu, Predicting networks-on-chip traffic congestion with spiking neural networks. Journal of Parallel and Distributed Computing, 154, (2021) 82-93. https://doi.org/10.1016/j.jpdc.2021.03.013

[15] N. Rohbani, Z. Shirmohammadi, M. Zare, S.-G. Miremadi, LAXY: A Location-Based Aging-Resilient Xy-Yx Routing Algorithm for Network on Chip. IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, IEEE, 36(10), (2017) 1725-1738. https://doi.org/10.1109/TCAD.2017.2648817

[16] K. Balamurugan, B.S. Rao, M. Vijayaraj, Performance Analysis of Optad-NoC: A Novel Optimized Routing Algorithm and Intelligent Router for 3D Network-on-Chip. Wireless Personal Communications, 121(4), (2021) 2511-2528. https://doi.org/10.1007/s11277-021-08834-0

[17] S. Srivastava, M. Moharir, S. Gunisetty, Performance analysis of congestion-aware Q-routing algorithm for network on chip. IAES International Journal of Artificial Intelligence, 13(1), (2024) 798-806. https://doi.org/10.11591/ijai.v13.i1.pp798-806

[18] M.I. Bondarenko, Overview of routing algorithms for network on chip. Journal Scientific and Technical of Information Technologies, Mechanics and Optics, 24(5), (2024) 687-698. https://doi.org/10.17586/2226-1494-2024-24-5-687-698

[19] R. Mahar, S. Choudhary, (2017) Performance evaluation of cross-link fully adaptive routing algorithm with cross-link architecture for Network on Chip. 2017 International Conference on Inventive Computing and Informatics (ICICI), IEEE, Coimbatore, India. https://doi.org/10.1109/ICICI.2017.8365198

[20] A.I. Fasiku, B.O. Ojedayo, O.E. Oyinloye, (2020) Effect of Routing Algorithm on Wireless Network-on-Chip Performance. 2020 Second International Sustainability and Resilience Conference Technology and Innovation in Building Designs (51154), IEEE, Sakheer, Bahrain. https://doi.org/10.1109/IEEECONF51154.2020.9319964

[21] M.T. Balakrishnan, T.G. Venkatesh, A.V. Bhaskar, Design and implementation of congestion aware router for network-on-chip. Integration, 88, (2023) 43-57. https://doi.org/10.1016/j.vlsi.2022.08.012

[22] N.S. Vindhya, B.M. Vidyavathi, (2018) Network on Chip: A Review of Fault Tolerant Adaptive Routing Algorithm, 2018 International Conference on Electrical, Electronics, Communication, Computer, and Optimization Techniques (ICEECCOT), IEEE, Msyuru, India. 1107-1110. https://doi.org/10.1109/ICEECCOT43722.2018.9001333

[23] P.C.D. Paris, E.C. Pedrino, A high-level simulator for Network-on-Chip. Integrated Computer-Aided Engineering, 32(1), (2025) 57-73. https://doi.org/10.3233/ICA-240743

[24] Y. Asadi, Optical network-on-chip (ONoC) architectures: a detailed analysis of optical router designs. Journal of Semiconductors, 46(3), (2025) 031401. https://doi.org/10.1088/1674-4926/24060006

[25] K. Khan, S. Pasricha, A Reinforcement Learning Framework with Region-Awareness and Shared Path Experience for Efficient Routing in Networks-on-Chip. IEEE Design & Test, IEEE, 40(6), (2023) 76-85. https://doi.org/10.1109/MDAT.2023.3306719

[26] F. Farahnakian, M. Ebrahimi, M. Daneshtalab, P. Liljeberg, J. Plosila, Bi-LCQ: A low-weight clustering-based Q-learning approach for NoCs. Microprocessors and Microsystems, 38(1), (2014) 64-75. https://doi.org/10.1016/j.micpro.2013.11.008

[27] A. Lit, J.S. Joshima, S.B. Suhaili, N. Rajaee, S.K. Sahari, R. Sapawi, Evaluation of Deterministic Routing on 100-cores Mesh Wireless NoC. 2023 International Conference on Artificial Intelligence in Information and Communication (ICAIIC), IEEE, Bali, Indonesia. (2023) 143-148. https://doi.org/10.09/ICAIIC57133.2023.10067074

[28] P.K. Sharma, P. Mitra, S. Biswas, A Deadlock-Free and Adaptive Prime Perspective Turn Model for 3D-Mesh Based Network-on-Chips, In: Woungang, I., Dhurandher, S.K., Pattanaik, K.K., Verma, A., Verma, P. (eds) Advanced Network Technologies and Intelligent Computing. ANTIC 2022. International Conference on Advanced Network Technologies and Intelligent Computing, Springer, Cham, 1797 (2022) 362-375. https://doi.org/10.1007/978-3-031-28180-8_24

[29] J. An, H. You, J. Sun, J. Cao, (2021) Fault tolerant xy-yx routing algorithm supporting backtracking strategy for noc, 2021 IEEE Intl Conf on Parallel & Distributed Processing with Applications, Big Data & Cloud Computing, Sustainable Computing & Communications, Social Computing & Networking (ISPA/BDCloud/SocialCom/SustainCom), IEEE, New York City, NY, USA. 632-635. https://doi.org/10.1109/ISPA-BDCloud-SocialCom-SustainCom52081.2021.00092

[30] M.B. Nejad, Parametric Evaluation of Routing Algorithms in Network on Chip Architecture. Computer Systems Science & Engineering,

35(5), (2020). https://doi.org/10.32604/csse.2020.35.367

[31] S.C. Lee, T.H. Han, Q-function-based traffic-and thermal-aware adaptive routing for 3D network-on-chip. Electronics, 9(3), (2020) 392. https://doi.org/10.3390/electronics9030392

[32] Y. Guo, Z. Huang, M. Ding, B. Lin, H. Luo, PROM: A persistent routing optimization method based on supervised learning. Journal of Network and Computer Applications, 242, (2025) 104223. https://doi.org/10.1016/j.jnca.2025.104223

[33] P. Dere, M. Gaikwad, S. Yadav, High-Performance Low-Power Inter- and Intra-Cluster CDQ-Routing for Cluster Diagonal Mesh. IETE Journal of Research, 71(10), (2025) 3337–3357. https://doi.org/10.1080/03772063.2025.2510520

[34] M. Ebrahimi, H. Tenhunen, M. Dehyadegari, Fuzzy-based adaptive routing algorithm for networks-on-chip. Journal of Systems Architecture, 59(7), (2013) 516-527. https://doi.org/10.1016/j.sysarc.2013.03.006

[35] J. Lavina, (2007). A simulator for NoC interconnect routing and application modeling. University of Southampton UK & Malaviya National Institute of Technology, UK & India.

[36] N. Ji, Y. Yang, A pre-congestion-aware deterministic-adaptive hybrid routing (PcaDAHR) algorithm for network-on-chip. Microelectronics Journal, 164, (2025) 106811. https://doi.org/10.1016/j.mejo.2025.106811

[37] M. Kaleem, I.F. Isnin, Thermal-aware directional and adaptive routing algorithm for 3D network-on-chip. Indonesian Journal of Electrical Engineering and Computer Science, 27(2), (2022) 1051-1061. https://doi.org/10.11591/ijeecs.v27.i2.pp1051-1061

[38] E. Taheri, A. Patooghy, K. Mohammadi, XYZ-ZXY: A minimal routing algorithm for dynamic thermal management in 3D NoCs. Iranian Conference on Electrical Engineering, IEEE, Shiraz, Iran. https://doi.org/10.1109/IranianCEE.2016.7585766

[39] K.A. Khan, S. Pasricha, (2023) A Reinforcement Learning Framework with Region-Awareness and Shared Path Experience for Efficient Routing in Networks-on-Chip, arXiv preprint. https://doi.org/10.48550/arxiv.2307.11712

[40] M. Manzoor, R.N. Mir, N.U.D. Hakim, PAAD (Partially adaptive and deterministic routing): A deadlock free congestion aware hybrid routing for 2D mesh network-on-chips. Microprocessors and Microsystems, 92, (2022) 104551. https://doi.org/10.1016/j.micpro.2022.104551

[41] J. Li, C. Qin, X. Sun, An efficient adaptive routing algorithm for the Co-optimization of fault tolerance and congestion awareness based on 3D NoC. Microelectronics Journal, 142, (2023) 105989. https://doi.org/10.1016/j.mejo.2023.105989

[42] A. Lit, M.H. Husin, S. Suhaili, Performance and energy evaluation of dynamic adaptive deterministic routing algorithm for multicore architectures. E-Prime - Advances in Electrical Engineering, Electronics and Energy, 9, (2024) 100716. https://doi.org/10.1016/j.prime.2024.100716

[43] A. Paul, S.P. Maity, Reinforcement learning based Q routing: performance evaluation on cognitive radio network topologies, Wireless Personal Communications, 125(2), (2022) 1425-1441. https://doi.org/10.1007/s11277-022-09612-2

[44] J. Jiao, R. Shen, L. Chen, J. Liu, D. Han, RLARA: A TSV-Aware Reinforcement Learning Assisted Fault-Tolerant Routing Algorithm for 3D Network-on-Chip. Electronics, 12(23), (2023) 4867. https://doi.org/10.3390/electronics12234867

Authors Contribution Statement

Prajakta P. Dere: Conceptualization, Methodology, Review and Editing, Shweta Ashtekar: Data Curation, Investigation, Experiments, Writing –Review and Editing. Ekta Sarda: Formal Analysis, Validation, Data Analysis, Methodology Development. Bhushan S. Deore: Original Draft Preparation, Software Implementation. All authors have read and agreed to the published version of the manuscript.

Funding

The authors declare that no funds, grants or any other support were received during the preparation of this manuscript.

Competing Interests

The authors declare that there are no conflicts of interest regarding the publication of this manuscript.

Data Availability

The data supporting the findings of this study can be obtained from the corresponding author upon reasonable request.

Has this article screened for similarity?

Yes

About the License