COMPARISON ANALYSIS OF DEADLOCK-FREE WORMHOLE ROUTING FOR INTERCONNECTED MULTIPROCESSOR NETWORK

Kamalesh Chandra Maurya[1] and Dr. Pankaj Kumar[2]

[1]*Research scholar, Dr. A.P.J. Abdul Kalam Technical University, Lucknow, kamalesh1947@gmail.com*

[2]*Professor, Department of Computer Science & Engineering, Shri Ramswaroop Memorial College of Engineering & Management Lucknow, pk79jan@gmail.com*

ABSTRACT:

Interconnected multiprocessor networks (IMNs) play a pivotal role in modern parallel computing systems, facilitating efficient communication among processors and minimizing latency. Wormhole routing has emerged as a prominent mechanism to manage data transfer in these networks, offering lower latency and higher bandwidth utilization compared to other routing techniques. However, wormhole routing is susceptible to deadlocks, a critical concern that can disrupt the overall system performance. This research presents a comprehensive comparison analysis of various deadlock-free wormhole routing algorithms used in interconnected multiprocessor networks. The objective is to evaluate their respective strengths and weaknesses, aiding system designers in making informed decisions when selecting the most suitable routing strategy for their specific application. The study commences with an in-depth review of deadlock scenarios and the challenges associated with wormhole routing. It delves into the theoretical foundations of deadlock avoidance and resolution algorithms, including virtual channels, path-based, and adaptive routing techniques. Each algorithm's mechanisms and overheads are analyzed to assess their suitability in diverse IMN topologies. To gauge the performance of these deadlock-free wormhole routing algorithms, a simulation framework is developed, encompassing various traffic patterns and network scales. Key metrics, such as, and network congestion, are employed to measure the algorithms' efficacy under both uniform and non-uniform traffic loads. The experimental results reveal the distinct advantages and limitations of each algorithm under specific scenarios. Some algorithms excel in uniform traffic environments, while others demonstrate superior performance in dealing with varying traffic patterns or large-scale network configurations.

Keywords: *Deadlock-free wormhole routing, multiprocessor network, latency, throughput, deadlock avoidance.*

1. INTRODUCTION

Interconnected Multiprocessor Networks (IMNs) have become the backbone of modern parallel computing systems, providing the essential communication infrastructure for high-performance computing applications. IMNs consist of multiple processing elements (PEs) interconnected through a network of links and routers, enabling efficient data exchange and cooperation among processors [2]. One critical aspect of IMNs is the routing mechanism employed to transmit data between PEs, as it directly impacts the overall system performance.

Wormhole routing has emerged as a popular technique for data transfer in IMNs due to its low latency and high bandwidth utilization. Unlike traditional store-and-forward techniques [16],

wormhole routing breaks data packets into smaller flits (flow control digits) and forwards them consecutively through intermediate routers towards their destination. This pipelined approach reduces queuing delays and improves overall communication performance.

However, wormhole routing is susceptible to a challenging problem known as deadlock[6]. Deadlock occurs when multiple PEs, each holding a flit and waiting to acquire additional resources, create a cyclic dependency[14], leading to a standstill in the network's progress. Deadlocks can significantly degrade system performance and, if left unresolved, may result in a complete system freeze.

To overcome this limitation, researchers have developed various deadlock-free wormhole routing algorithms. These algorithms employ specific strategies to ensure that deadlocks are avoided or resolved if they occur, without sacrificing the benefits of wormhole routing. The primary objective of these algorithms is to maintain a high level of network performance while guaranteeing deadlock-free operation.

This research aims to conduct a comprehensive comparison analysis of different deadlock-free wormhole routing algorithms for interconnected multiprocessor networks. By evaluating the strengths and weaknesses of these algorithms, we seek to provide valuable insights for system designers and researchers to make informed decisions when selecting the most suitable routing strategy for their specific application scenarios.

The remainder of this study is organized as follows: Section 2 provides a literature review on wormhole routing and the challenges associated with deadlock. Section 3 outlines the theoretical foundations of deadlock avoidance and resolution algorithms. Section 4 presents the experimental methodology and simulation framework used to evaluate the algorithms' performance. Section 5 discusses the experimental results and their implications. Finally, Section 6 concludes the research and highlights its contributions to the field of IMNs and parallel computing systems.

2. LITERATURE REVIEW

In first generation mesh NoCs, Dimension Order Routing (DOR), such as XY routing, has been a popular option. Adaptive routing solutions offer greater performance and fault tolerance by providing alternate pathways, even if XY routing is widely popular because to its simplicity [18, 23]. The two partially adaptive deadlock-free routing algorithms that are most frequently employed are the turn models and odd-even models. In some situations, none of these two routing approaches is strictly efficient. In some circumstances, non-minimal routing results in livelocks [23]. Because the level of adaptability is uniformly distributed over the network, the odd-even adaptive routing model is preferred over other turn models. Based on network congestion circumstances, the DyAD smart routing [23] effectively alternates between adaptive and deterministic routing.

J. Duato has proposed a necessary and sufficient condition for deadlock-free adaptive routing in wormhole-switched networks [15]. In his paper titled "A Necessary and Sufficient Condition for Deadlock-Free Adaptive Routing in Wormhole Networks," Duato presented a comprehensive analysis of deadlock-free routing in wormhole-switched networks. He introduced a condition

called "turn model" and proved that it is both necessary and sufficient for deadlock avoidance in adaptive routing algorithms.

Jindun Dai [3] presented a novel deadlock-free adaptive routing algorithm for 3D mesh NoC interconnections. The routing rules of traditional XY routing and YX routing are relaxed and used for intra-layer routing.

Yuan Cai [1] proposed a new 3D Network-on-Chip(NoC) routing algorithms aim to efficiently utilize the vertical connections in a vertically stacked 2D mesh topology to improve communication performance and reduce congestion. These algorithms take advantage of the additional routing paths provided by the vertical connections.

Congestion Aware Deterministic Routing (CADR) [22] puts out a practical, affordable way to gauge network congestion. They construct optimal routing pathways for all trace flows based on this estimate. This method is deterministic and works best on reconfigurable systems that run several apps that frequently do repeated calculations on huge amounts of data. Furthermore, CADR's performance isn't much better than that of the DyAD model. A load balancing routing technique called Path- Based Randomised Oblivious Minimal Routing (PROM) [24] examines the path variety in routes. PROM uses randomness to provide load balancing. Even though randomization has less overhead, a random selection strategy could not match the performance of an adaptive routing scheme with adequately assessed congestion frame- work.

Both Glass and Ni have proposed methodologies for generating deadlock-free routing algorithms. Both proof techniques require an acyclic channel dependency graph. Glass and Ni propose a method of analyzing routing algorithms based on the permitted and prohibited dependencies [18] from one channel to another. Boura and Das propose a method of proving deadlock freedom [17] by partitioning the channels into two acyclic sets and requiring messages to route completely in the first set before using channels in the second set.

Kawano, et al design a new routing method based on HiRy that can increase the number of permitted paths and thus can improve the network performance [21]. To support all source-and destination pairs reachable and to reduce the average path length, a heuristic approach is introduced.

G M. Chiu presented the Odd-Even turn model and a deadlock-free adaptive routing algorithm for mesh networks [13]. The Odd-Even turn model is a routing technique that uses a combination of odd and even turns to avoid deadlocks in 2D mesh networks. It guarantees deadlock freedom by ensuring that packets always move in a direction that eliminates potential cyclic dependencies between channels.

Freek Verbeek proposed an algorithm that automatically proves routing functions deadlock-free or outputs a minimal counter-example explaining the source of the deadlock [6].

Wei Luo [8] discussed one commonly used adaptive deadlock-free routing algorithm for torus networks is the Dimension-Ordered Routing (DOR) algorithm. DOR ensures deadlock avoidance

by carefully selecting the next hop for each packet based on the current and destination coordinates in the torus network.

3. THEORETICAL FOUNDATIONS OF DEADLOCK AVOIDANCE AND RESOLUTION ALGORITHMS

Deadlock is a critical issue in concurrent computing systems, where multiple processes compete for resources and can get stuck in a situation where each process is waiting for a resource that is held by another process. Deadlock avoidance and resolution algorithms are designed to prevent or resolve deadlocks in such systems.

Deadlock avoidance and resolution strategies in the context of wormhole switching systems share similarities with those in general concurrent computing systems, but there are specific considerations due to the nature of wormhole switching. Wormhole switching is a technique used in computer networks and parallel computing systems to efficiently transmit data between nodes.

3.1 DEADLOCK AVOIDANCE IN WORMHOLE SWITCHING

In wormhole switching, messages are divided into flits (flow control digits) and transmitted through the network using virtual channels. Each virtual channel has its own buffer, and deadlock can occur if multiple flits contend for the same virtual channel and are unable to progress due to resource conflicts.

To avoid deadlocks in wormhole switching, designers can employ various strategies:

a) **Virtual Channel Allocation:** Ensuring that each flit has a dedicated virtual channel can help avoid contention and reduce the chances of deadlock.
b) **Non-Blocking Routing:** Employing non-blocking or adaptive routing algorithms can prevent the formation of deadlock-causing cycles.
c) **Minimal Adaptive Routing:** Routing algorithms that minimize the number of adaptive decisions can help reduce the complexity of the system and avoid potential deadlocks.
d) **Virtual Channel Priority:** Assigning priorities to virtual channels can help resolve contention conflicts and improve overall system performance.

3.2 DEADLOCK DETECTION AND RESOLUTION IN WORMHOLE SWITCHING

In some cases, deadlock avoidance techniques may not be sufficient or may not be applicable due to system constraints. In such situations, deadlock detection and resolution become necessary. However, deadlock detection in wormhole switching can be complex and resource-intensive.

a) **Local Deadlock Detection:** Wormhole switching systems can employ local deadlock detection algorithms at individual switches or nodes to identify potential deadlock situations. Each switch monitors its internal state and the state of its connected links.
b) **Distributed Deadlock Detection:** In more complex wormhole switching networks, distributed deadlock detection algorithms can be used, where switches collaborate to detect deadlocks in the entire network.

c) **Deadlock Resolution:** Once a deadlock is detected, there are several strategies that can be employed for deadlock resolution. These include route reversal, priority inversion, and flit preemption. In route reversal, the flits involved in the deadlock may be directed back along their paths to break the deadlock. Priority inversion involves adjusting the priorities of conflicting flits. Flit preemption allows the system to interrupt and reroute certain flits to resolve the deadlock.

It's essential to carefully design and implement deadlock avoidance and resolution mechanisms in wormhole switching systems to ensure efficient data transmission and prevent potential performance bottlenecks due to deadlocks. These strategies must be tailored to the specific network topology and traffic patterns in the system.

3.3 DEADLOCK-FREE ROUTING ALGORITHMS

There are several deadlock-free routing algorithms that have been developed to prevent deadlocks in different network architectures. Here we discuss some of the main deadlock-free routing algorithm:

(a) Dimension-Ordered Routing:

Dimension-Ordered Routing is a type of routing algorithm used in parallel computing systems with interconnection networks arranged in multiple dimensions. In such systems, nodes are connected in a structured manner, forming a mesh, torus, or hypercube network topology. Each dimension of the network represents a specific direction or path along which data can be transmitted. While Dimension-Ordered Routing is simple to implement and guarantees deadlock-free routing, it may not always be the most efficient choice, especially when there are alternative paths with less contention or congestion

The basic idea behind DOR routing is to route messages in a deterministic and ordered manner by following a specific dimension order. Messages are routed along each dimension one at a time until they reach their destination. The order in which the dimensions are traversed is predetermined and consistent across all nodes in the network. **XY routing** is the best known example of DOR routing. Figure 1 shows out of 8 possible turn only 4 turns are allowed in XY routing or YX routing algorithm.

(a) (b)

Figure1: Turn allowed (a) XY routing and (b) YX routing

XY Routing Algorithm:

XY routing algorithm comes under deterministic routing algorithm. This algorithm can be implemented for both for regular and irregular network topology. It is called dimension order

routing (DOR). It follows the concept of minimal turning routing. In this routing each node or router of NoC is identified by the (x, y) co-ordinates of that node for a 2D mesh. According to this algorithm the data packets will traverse in X-direction towards the destination column. After finding the destination column the data packets will traverse to the destination node. This algorithm simply states that "First the data will move in X-direction and then in Y-direction". That is why the name of the algorithm is XY Routing algorithm. According to this algorithm the packets can't move first in Y-direction then in X-direction. So it has some routing or turning restrictions. Due to which it becomes deadlock free. This algorithm will be chooses in the condition when the number of column is more than number of row in the mesh network.

YX Routing Algorithm:

YX routing algorithm similar to XY routing algorithm, but in this algorithm first the data will move in Y-direction and then in X-direction. According to this algorithm the packets can't move first in X-direction then in Y-direction. So it has some routing or turning restrictions. Due to which it becomes deadlock free.

(b) Adaptive Routing:

Adaptive Routing[13] is a dynamic routing approach in which the selection of the path for data transmission is made based on the current network conditions. Unlike static routing algorithms, which use fixed paths, adaptive routing algorithms consider factors like network congestion, link failures, and other performance metrics to make real-time decisions about the best path to take.
In adaptive routing, the routing decision is often taken by the switches or routers in the network. These devices monitor the network state and make routing decisions on a per-packet basis. This flexibility allows adaptive routing to adapt to changes in the network and improve overall performance and fault tolerance[9].

The main advantage of adaptive routing is that it can effectively utilize available network resources and avoid congested or faulty links. However, the complexity of adaptive routing algorithms and the need for frequent updates can make their implementation more challenging than simpler routing methods.

4. EXPERIMENTAL METHODOLOGY AND SIMULATION

We are using NIRGAM 2.1 simulator. NIRGAM is an extensible and modular SystemC based simulator (NIRGAM), which let the user plug-in and experiment with different applications and routing algorithms. It allows the user to analyze the performance (Average latency, throughput and total network power) of a NoC design for a user specified application and a user specified routing algorithm. At present, NIRGAM (NoC Interconnect RoutinG and Applications' Modeling) simulator supports mesh, torus, mesh with link failures and irregular topologies with wormhole switching mechanism.
The parameters of the nirgam simulator describe in Table1. The experimental setup for calculating the performance of various routing algorithms for different dimensions of mesh

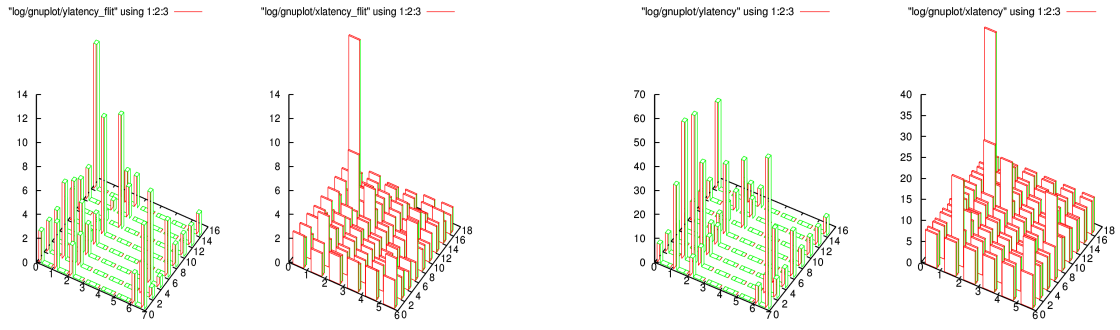
topology, with each node linked to a traffic generator which produces CBR (Constant Bit Rate) of the value 2 Gbps for every pair of source and destination. We take input channel FIFO buffer depth (number of buffers) is 8 and each physical channel has two virtual channels. The clock frequency of 1 GHz and packet size is 32 bytes, with a flit interval of 2 clock cycle.

TABLE 1: Parameters of the NIRGAM Simulator

Parameter Name	**Description (Value)**
TOPOLOGY	Defines 2-dimensional **Mesh** / Torus topology.
NUM_ROWS	Defines number of rows in the selected topology. (9)
NUM_COLS	Defines number of columns in the selected topology. (9)
NUM_BUFS	Number of buffers in input channel fifo. (8)
RT_ALGO	Name of routing algorithm. (XY and YX)
FLITSIZE	Size of flit in bytes. (2 bytes)
DIRNAME	Directory name in which results will be stored after simulation
LOG	Defines log level for the event log generated
WARMUP	Defines warm-up period: number of clock cycles before traffic generation begins. (1000 clock cycles)
SIM_NUM	Defines clock cycles for which simulation runs. (10000 clock cycles)
TG_NUM	Defines clock cycle until which traffic is generated.(7000 clock cycles)
FLIT_INTERVAL	Interval between successive flits in clock cycles. (2 clock cycles)
LOAD	Percentage load. This determines percentage of maximum bandwidth being used. (100%)
PKT_SIZE	Packet size in bytes. (32 bytes)
CLK_ FREQ	Defines clock frequency in GHz. (1 GHz)
DESTINATION	Destination tileID, may be Fixed or Random

5. EXPERIMENTAL RESULTS

The experiments are performed on NIRGAM simulator for various combinations of source tile (node) for XY and YX routing algorithms. We use size of 9x9 2-dimensional mesh topology. After running simulator the screenshots of result shown in figures. Figure 2(a) shows Overall Average Latency per channel (in clock cycles per flit), while figure 2(b) shows Overall Average Latency per channel (in clock cycles per Packet), figure 3 shows Average Throughput (Gbps) and figure 4 shows Power consumption at nodes (in watt).



(a) *(b)*

***Figure 2**. (a) Overall Average Latency per channel (in clock cycles per flit)
(b) Overall Average Latency per channel (in clock cycles per Packet)*

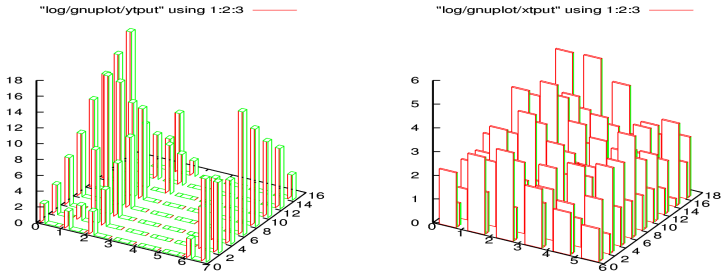


***Figure 3**. Average Throughput (Gbps)*

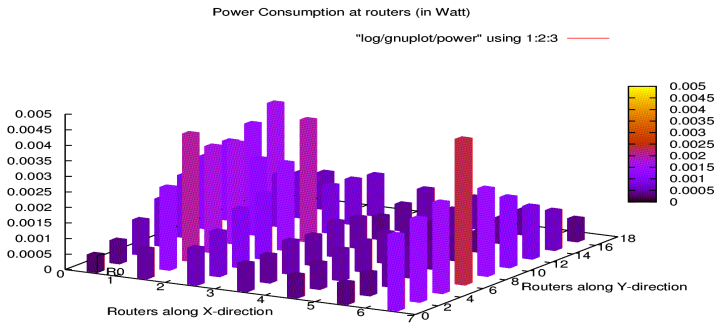


***Figure 4**. Power consumption at nodes (in watt)*

TABLE 2: Overall average latency of XY and YX routing algorithm

S No.	Random traffic pattern of source node	Overall average latency per channel (in clock cycles per flit)		Overall average latency per channel(in clock cycles per packet)		Overall average latency (in clock cycles per flit)	
		XY	YX	XY	YX	XY	YX
1.	0 CBR.so	1.56239	1.56239	14.0615	14.0615	125.944	125.944
2.	0 CBR.so 1 CBR.so	1.62372	1.62372	14.6135	14.6135	116.273	116.273
3.	0 CBR.so 2 CBR.so	1.54796	1.56239	13.9316	14.0615	124.758	125.944
4.	1 CBR.so 2 CBR.so	1.61816	1.64658	14.5635	14.8192	115.873	117.944
5.	0 CBR.so 3 CBR.so	1.43947	1.43994	12.9552	12.9594	141.895	141.944
6.	0 CBR.so 4 CBR.so	1.4835	1.49504	13.3515	13.4554	132.891	133.944
7.	0 CBR.so 9 CBR.so	1.99099	1.99099	17.9189	17.9189	97.9436	97.9436
8.	0 CBR.so 10 CBR.so	1.41598	1.41598	12.7438	12.7438	145.944	145.944
9.	0 CBR.so 1 CBR.so 2 CBR.so	1.72955	1.60164	15.5659	14.4148	129.765	119.496
10.	0 CBR.so 1 CBR.so 3 CBR.so	1.6334	1.50333	14.7006	13.53	140.946	130.163
11.	0 CBR.so 4 CBR.so 10 CBR.so	1.41418	1.42373	12.7276	12.8136	143.627	144.61
12.	0 CBR.so 9 CBR.so 10 CBR.so	1.61627	1.61627	14.5465	14.5465	120.61	120.61
13.	0 CBR.so 1 CBR.so 2 CBR.so 3 CBR.so	1.93601	1.51732	17.4241	13.6559	156.554	129.108
14.	0 CBR.so 1 CBR.so 9 CBR.so 10 CBR.so	1.63515	1.63515	14.7163	14.7163	117.108	117.108

15.	0 CBR.so 1 CBR.so 2 CBR.so 9 CBR.so 10 CBR.so	1.69425	1.61918	15.2483	14.5726	124.791	118.875
16.	0 CBR.so 1 CBR.so 2 CBR.so 3 CBR.so 4 CBR.so	2.07767	1.49995	18.699	13.4995	177.66	131.675
17.	0 CBR.so 3 CBR.so 4 CBR.so 9 CBR.so 10 CBR.so	1.61825	1.49866	14.5642	13.4879	142.409	131.571
18.	0 CBR.so 1 CBR.so 2 CBR.so 3 CBR.so 4 CBR.so 10 CBR.so	1.80881	1.45356	16.2793	13.0821	174.559	136.743
19.	0 CBR.so 1 CBR.so 2 CBR.so 3 CBR.so 4 CBR.so 9 CBR.so 10 CBR.so	1.91014	1.52944	17.1913	13.765	153.7	127.2

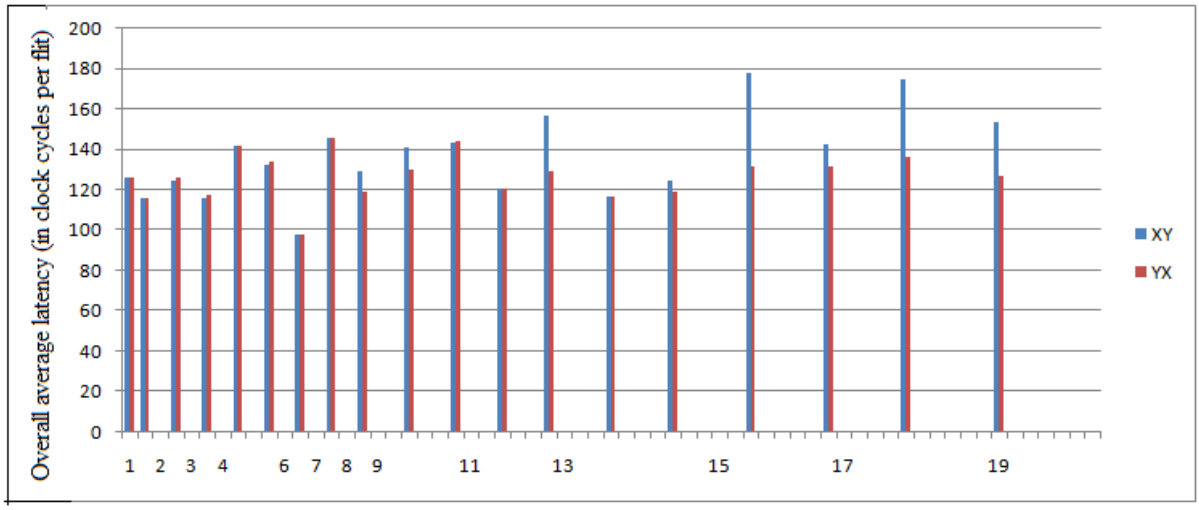


***Figure 5**. Overall average latency of XY and YX routing algorithm*

The simulation results are evaluated for Overall Average Latency per channel (clk cycle/packet) for 9x9 mesh network in uniform random traffic pattern. In uniform random traffic pattern source node send packets to all remaining nodes of network. Table 2 or figure 5 shows the comparative performance of XY and YX routing algorithm for some particular traffic pattern. Result shows that if three or more source nodes are present in the same row the YX routing is better performing than XY routing.

6. CONCLUSION AND DISCUSSION

In conclusion, this research contributes an insightful comparison analysis of deadlock-free wormhole routing algorithms for interconnected multiprocessor networks. The findings offer valuable guidance to system designers and researchers, aiding them in selecting the most suitable routing approach based on their specific IMN requirements. Ultimately, the study aims to enhance the overall efficiency and reliability of parallel computing systems through informed routing decisions.

From results of experiments performed on NIRGAM simulator we can conclude that any one of the algorithm does not perform always better. For some traffic pattern XY routing perform better while for some traffic pattern YX routing perform better. It is also concluded that if three or more source nodes are present in the same row then YX routing is better than XY routing, while if three or more source nodes are present in the same column then XY routing is better than YX routing.

Furthermore, the study explores the trade-offs between latency, energy consumption, and network resource utilization in the context of deadlock-free wormhole routing. Additionally, fault tolerance and adaptability to dynamic traffic conditions are considered to provide a holistic understanding of the algorithms' capabilities.

References:

1. Yuan Cai , Dong Xiang and Xiang Ji, “Deadlock-Free Adaptive Routing Based on the Repetitive Turn Model for 3D Network-on-Chip”, *IEEE Intl Conf on Parallel & Distributed Processing with Applications, Dec. 2018*
2. Yoon Seok Yang , Hrishikesh Deshpande , Gwan Choi and Paul V. Gratz “SDPR: Improving Latency and Bandwidth in On-Chip Interconnect Through Simultaneous Dual-Path Routing”, *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, Vol: 37 , No. 3 , pp. 545 – 558, March 2018*.
3. Jindun Dai, Xin Jiang, Renjie Li and Takahiro Watanabe “An Efficient Deadlock-Free Adaptive Routing Algorithm for 3D Network-on-Chips”, *IEEE 11th International Symposium on Embedded Multicore/Many-core Systems-on-Chip (MCSoC), pp. 29 – 36, 2017*
4. Vahid Janfaza and Elaheh Baharlouei “A new fault-tolerant deadlock-free fully adaptive routing in NOC”, *IEEE East-West Design & Test Symposium (EWDTS), pp 1-6, 2017*
5. Canwen Xiao, Yue Yang, and Jianwen Zhu ”A Sufficient Condition for Deadlock-Free Adaptive Routing in Mesh Networks”, *IEEE Computer Architecture Letters, Vol. 14, No. 2, July-December 2015*
6. Freek Verbeek and Julien Schmaltz” A Decision Procedure for Deadlock-Free Routing in Wormhole Networks,” *IEEE Transactions On Parallel And Distributed Systems Vol. 25, No. 8, August 2014.*
7. Mark Karpovsky, Lev Levitin and Mehmet Mustafa “Optimal Turn Prohibition for Deadlock Prevention in Networks with Regular Topologies,” *Transactions On Control Of Network Systems, Vol. 1, No. 1, March 2014*

8. Wei Luo and Dong Xiang, "An Efficient Adaptive Deadlock-Free Routing Algorithm for Torus Networks" IEEE TRANSACTIONS ON PARALLEL AND DISTRIBUTED SYSTEMS, VOL. 23, NO. 5, MAY 2012
9. D. Xiang, "Deadlock-free Adaptive Routing in Meshes with Fault-Tolerance Ability Based on Channel Overlapping," *IEEE Trans. Dependable and Secure Computing, vol. 8, no. 1, pp. 74-88, Jan./Feb. 2011.*
10. D. Xiang, Y. Pan, Q. Wang, and Z. Chen, "Deadlock-Free Fully Adaptive Routing in 2-Dimensional Tori Based on New Virtual Network Partitioning Scheme," *Proc. 28th Int'l Conf. Distributed Computing Systems, pp. 454-461, June 2008.*
11. Singh, W.J. Dally, A.K. Gupta, and B. Towles, "Goal: A Load-Balanced Adaptive routing Algorithm for Torus Networks," *SIGARCH Computer Architecture News, vol. 31, no. 2, pp. 194-205, 2003.*
12. J. Duato and T.M. Pinkston, "A General Theory for Deadlock-Free Adaptive Routing Using a Mixed Set of Resources," *IEEE Trans. Parallel Distributed Systems, vol. 12, no. 12, pp. 1219-1235, Dec. 2001.*
13. G M. Chiu, "The Odd-Even Turn Model for Adaptive Routing," *IEEE Trans. Parallel and Distributed Systems, vol. 11, no. 7, pp. 729-738, July 2000.*
14. Rajendra V. Boppana and Suresh Chalasani "A Framework for Designing Deadlock-Free Wormhole Routing Algorithms" *IEEE Transactions On Parallel And Distributed Systems, Vol. 7, No. 2, February 1996.*
15. J. Duato, "A Necessary and Sufficient Condition for Deadlock-Free Adaptive Routing in Wormhole Networks," *IEEE Trans. Parallel Distributed Systems, vol. 6, no. 10, pp. 1055-1067, Oct. 1995.*
16. L.M. Ni and P.K. McKinley, "A Survey of Wormhole Routing Techniques in Direct Networks," IEEE Computer, Vol. 26, pp. 62-76, 1993
17. Y.M. Boura and C.R. Das, "A Class of Partially Adaptive Routing Algorithms for n-dimensional Meshes," *Proc. Int'l Conf. Parallel Processing, vol. III, pp. 175-182, Aug. 1993.*
18. C. Glass and L.M. Ni, "The Turn Model for Adaptive Routing," *Proc. 19th Ann. Int'l Symp. Computer Architecture, pp. 278-287, May 1992.*
19. J. Duato, "On the Design of Deadlock-Free Adaptive Routing Algorithms for Multicomputers: Design Methodologies," *Proc. Parallel Architectures and Languages Europe 91, pp. 390-405, June 1991.*
20. W.J. Dally and C.L. Seitz, "Deadlock-Free Message Routing in Multiprocessor Interconnection Networks," IEEE Trans. Computers, vol. C-36, no. 5, pp. 547-553, May 1987.
21. Kawano et al."A Generalized Theory Based On The Turn Model For Deadlock-Free Irregular Networks", IEICE TRANS. INF. & SYST., Vol.E103–D, No.1 January 2020
22. A. E. Kiasari, A. Jantsch, and Z. Lu. "A framework for designing congestion-aware deterministic routing", NoCArc-'10: Proceedings of the 3rd International Workshop on Networks-on-Chip Architectures, pages 45-50, December 2010.
23. J. Hu and R. Marculescu. "DyAD: Smart routing for networks-on-chip", DAC-'04: Proceedings of the Design Automation Conference, pages 260-264, 2004.
24. M. H. Cho, M. Lis, K. S. Shim, M. Kinsy, and S. Devadas "Path-Based, Randomized, blivious, Minimal Routing" NoCArc-'09: Proceedings of the 2nd International Workshop on Networks-on-Chip Architectures, pages 23-28, December 2009.
25. Kamalesh Chandra Maurya and Pankaj Kumar Devadas "Partially adaptive deadlock-free routing algorithm for rectangular 2-d mesh network", Eur. Chem. Bull. 2023,12(10), pages 6058-6070