EE5902 Project

Evaluation of 2D Network-on-Chip Routing Algorithm

Group Number: 28

Group Member:

Ma Junting (A0144294A)

Wang Yawen (A0206856X)

Program files can be cloned from: https://github.com/majunting/EE5902_project

Project Overview

Network-on-chip (NOC) means that a silicon chip is used to carry out communication features for large-scale systems. The chips make routing decisions and forward the packets to the selected links. Messages can flow from the source module to the destination module via these links.

This project aims to evaluate the performance difference of routing algorithms for 2D NoC systems. Performance metrics used for the evaluation includes throughput, delay and power consumption. For a user of network-on-chip systems, throughput and power consumption are the most crucial parameters to be considered. Throughput refers to the rate of packets flowing through the system, a higher throughput refers to a higher packet transmission rate. Power consumption is important if the system is to be powered on and operated for a long period of time. Power consumption includes static and dynamic power. Static power refers to the power consumed for the setup of the network size, including the implementation of different routing algorithm. Different routing algorithm requires the chip to perform different computation, resulting in different static power consumption. On the other hand, dynamic power is the power used for packet transmission. For the same period of time, a higher throughput can result in a higher dynamic power consumption. In a network-on-chip system, it is possible for deadlock to happen. Deadlock refers to a situation where a bunch of packets hold on to the resources that other packets are waiting for, including the link or buffer, causing no packet to pass through. This can be indicated by the delay rate as well as the maximum delay cycles. For a packet that is delayed for a significantly large number of cycles, it is reasonable to conclude that it experiences a deadlock. Therefore, a lower delay rate is generally desirable.

Overview on Routing Algorithms

Routing algorithm determines the sequence of channels that packets traverse through the network. For NOC systems, there are three types of routing algorithms, including deterministic routing, adaptive routing and partially adaptive routing.

Deterministic routing algorithm routes packets from one point to the other along a fixed path, without considering the condition of the network. Dimension-order-routing and XY routing are the two popular deterministic routing algorithms implemented by NOC systems. In this project, XY routing is selected as the representation. XY routing algorithm does not check if the network is congested or not, and determines the forwarding direction of the packet.

Adaptive routing algorithm takes into account the topology and the condition of network traffic when deciding the path of a packet. The selected algorithm is Dynamic Adaptive Deterministic (DyAD) routing algorithm. It changes between different styles of routing based on the network traffic conditions. If the network is not congested, Odd-Even algorithm is implemented. On the other hand, if congestion is detected in the network, routing will be modified to accommodate the congestion.

The third type is partially adaptive routing algorithm, as these routing algorithms have limited adaptivity. Multiple routing algorithms are evaluated in this project, including West-First routing, North-Last routing, Negative-First routing and Odd-Even routing. These algorithms are relatively popular among the real-world NOC systems, and it would be interesting to examine the difference among these algorithms in terms of their performance.

For West-First, North-Last and Negative-First routing algorithms are turn restriction algorithms, which means these algorithms restricts some turn directions when forwarding packets. West-First routing algorithm restricts turns to west, which is negative x-direction, and any necessary west direction needs to be processed first. After processing the west direction movements, the routing algorithm works similarly as a XY routing algorithm. North-Last routing algorithm restricts turns if the current direction is north. Movements to north direction should be processed last. Negative-First algorithm restricts turns to negative directions. In x-direction, west is considered negative, and in y-direction, south is considered negative. The necessary negative turns need to be processed first before the other turns. Odd-Even routing algorithm has a better adaptivity compared to the other three partially adaptive turn restricting routing algorithms, as it restricts different turn directions in odd and even cycles. Odd-Even routing algorithm allows a better prevention of deadlock situations.

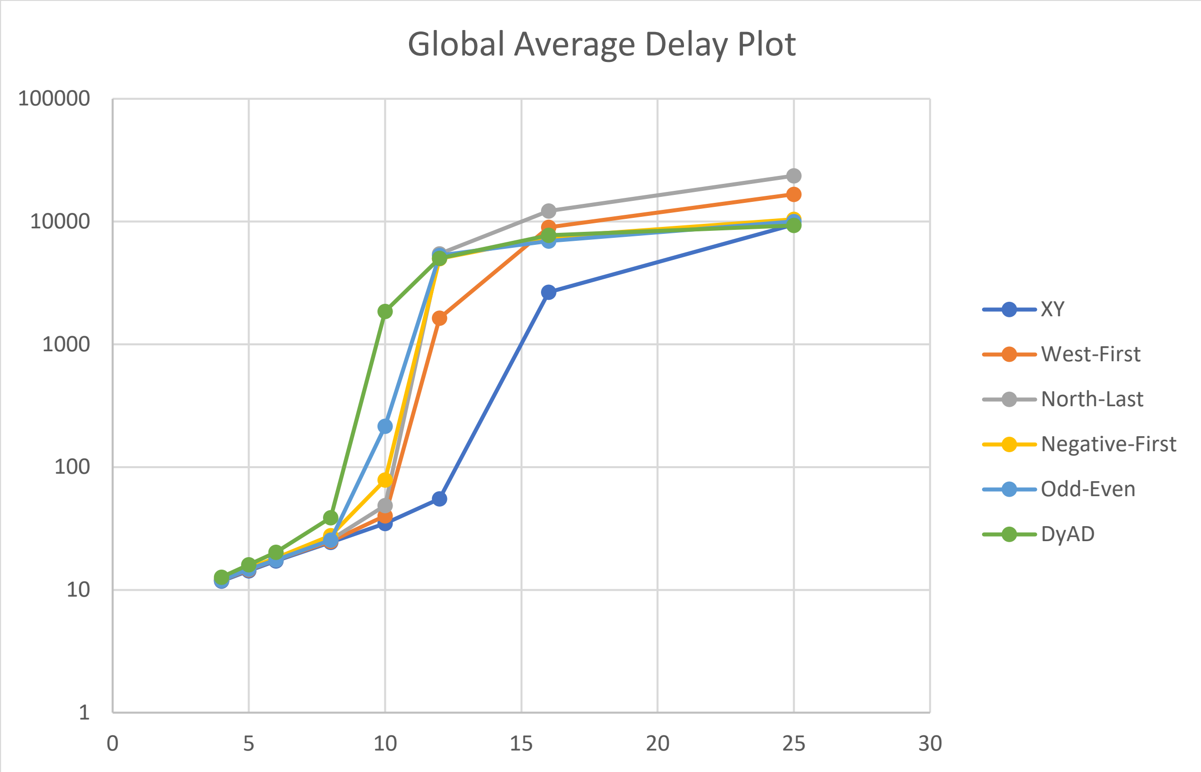
Simulation Tool and Setup

The simulation software used for this project is noxim, which is a NOC simulation software based on C++. Simulation parameters can be loaded into the command line interface of Noxim to set up the simulation environment. The parameters are saved in a YAML file, and these parameters including the network topology type and size, buffer size, the routing algorithm to be implemented and so on.

In the simulation process, due to the time constraint, the focus is on mesh topology and how mesh size affects performance of routing algorithms. In Noxim, the network topology is mesh topology, which means it has 2 dimensions: x-direction and y-direction. In the simulation setup, x-dimension and y-dimension are set to be the same for ease of comparison and more obvious indication of effect of network size. The effect of different x-dimension and y-dimension is, although may be an important factor on network performance, not taken into consideration.

For each simulation, the simulation time is 100,000 cycles to ensure a more accurate simulation result. Different mesh sizes, from 4 x 4 up to 25 x 25 are carried out in order to observe the effect more obviously. For each simulation setup and each routing algorithm, multiple iterations are carried out to ensure the results obtained is less affected by any possible random errors. For mesh size 25 x 25, simulation is done once for each routing algorithm, due to the significant increased operating time. The complete result data file has been submitted together with the report.

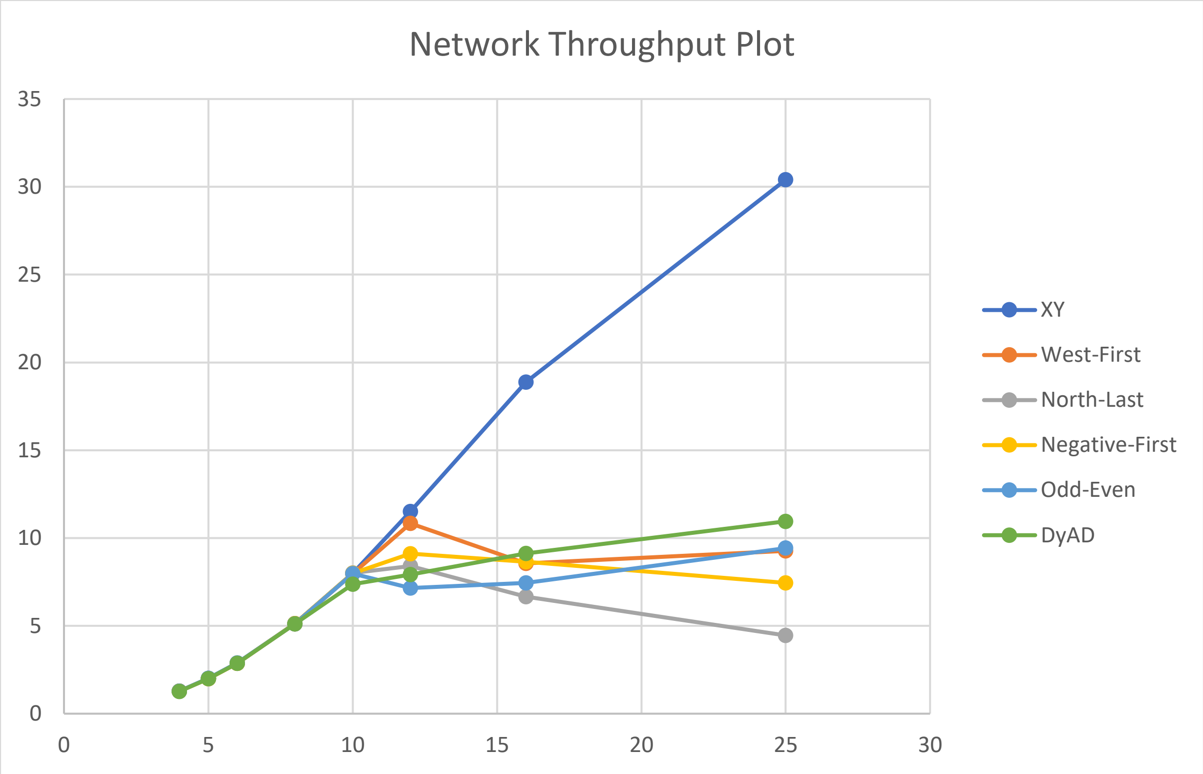
Simulation Result and Analysis



The figure above indicates the global average delay for different routing algorithm

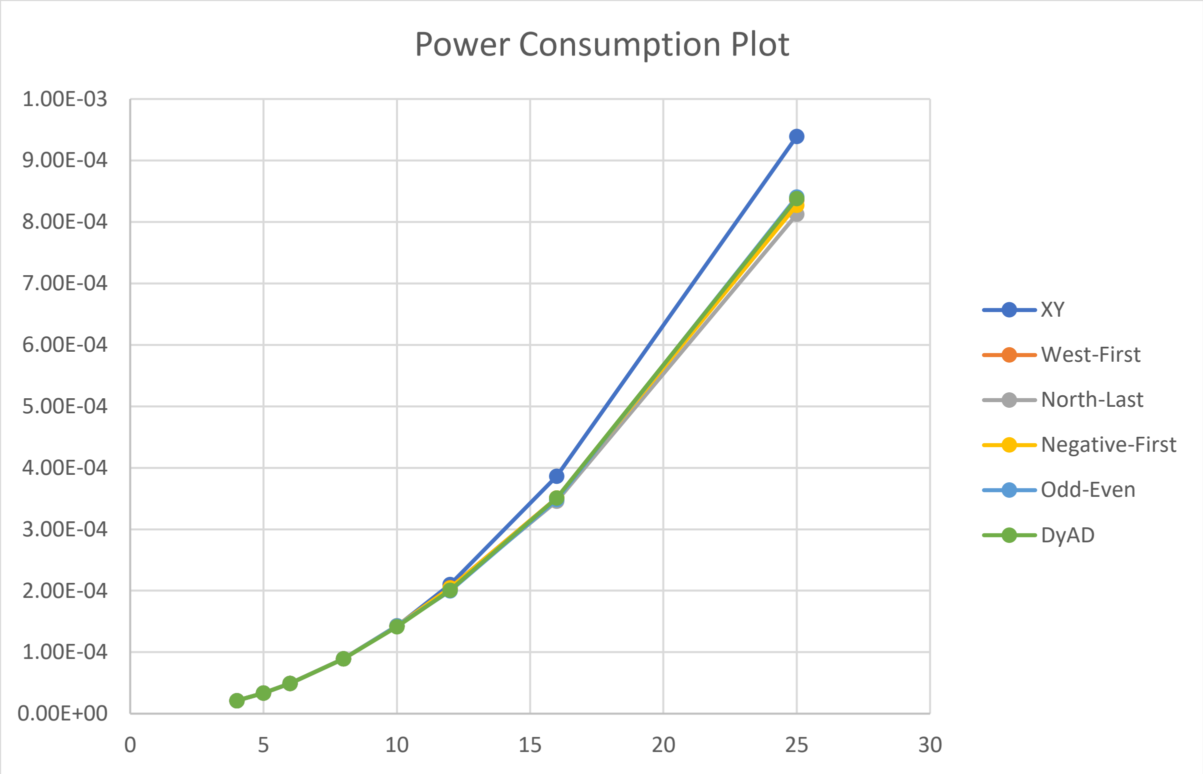
- Global delay increases significantly as the network mesh size increases. Due to the increase in network mesh size, the distance between source and destination increases, resulting in a longer time required for a packet to traverse through.
- For a very small mesh size (up to mesh size 6 x 6),
- The global average delay increases to the most extent when the mesh size is around 10 x 10. This can be seen as a threshold, after which the increase in global average delay becomes less significant.
- For DyAD which is the fully adaptive routing algorithm, the global average delay is greater than the other partially adaptive routing algorithms. This is reasonable as a more adaptive routing algorithm can lead to more computation, resulting in greater delay. When the network mesh size is greater than 10 x 10, it can be seen as a large network. DyAD routing algorithm is more advantageous by avoiding deadlocks and reduce global average delay.
- For a very large mesh size, turn restriction routing algorithms (to be more specific, North-Last and West-First) perform very badly in terms of global average delay, possibly due to the occurrence of deadlocks.

- Surprisingly XY routing algorithm has a great performance in global average delay, as it has a lower global average delay compared to all the other routing in almost all the mesh sizes.



The figure above shows the plot of network throughput against different mesh sizes

- Network throughput increases as the mesh size increases when mesh size is small. When mesh size reaches a specific value, in this case around 10 to 12, the network throughput does not have visible increase. This is possible to be because of the huge increase in global average delay around this mesh size.
- For fully adaptive DyAD routing algorithm, it performs better as the mesh size is very large compared to the other partially adaptive routing algorithms. It keeps its increasing trend in network throughput even at a very large mesh size.
- North-Last and Negative-First routing algorithms have very low network throughput when the network mesh size is very large. The poor performance of North-Last routing algorithm is reasonable from the global average delay performance. However, Negative-First routing algorithm performs okay in terms of global average delay. This indicates that global average delay does not directly relate to network throughput.
- XY routing algorithm’s network throughput increases linearly as the mesh size increases. This contradicts the theoretical analysis on the performance of XY routing algorithm and the other adaptive and partially adaptive routing algorithms.



This figure shows power consumption of different routing algorithms against mesh size

- As mesh size increases, power consumption increases as well, and the network throughput, which directly relates to the number of packets transmitted, contributes to the power consumption as well, but the effect is less significant compared to mesh size.
- A more adaptive routing algorithm requires a larger static power consumption compared to a more deterministic routing algorithm.

Discussion and Conclusion

- North-Last, West-First and Negative-First are all turn restricting routing algorithms. Theoretically they should have a similar performance in terms of global average delay and network throughput. However, the simulation results show otherwise. This could be because of the limitation of the simulation software. It is possible that Noxim always injects the packets in the similar manner, which means they generally requires turning to the same directions, or the pattern of packet injection appears cyclically.
- XY routing algorithm performs extremely well compared to all the other routing algorithms when the mesh size is large. For example, the network throughput of XY routing algorithm is three to four times greater than that of the other routing algorithms. Similar to the previous point, this could be due to the limitation of the simulation software.
- Due to time constraint, only square shape mesh network is simulated. For network-on-chip systems, the topology includes mesh network, ring network, torus network, binary tree network, spidergon network and application specific topology. Future work could be done on different network topologies, to get a better insight on the performance of different routing algorithms.

In conclusion, fully adaptive routing algorithm performs better as compared to the other partially adaptive routing algorithms when the network mesh size is large, but does not perform as good compared to the other partially adaptive routing algorithms when the mesh size is not very large. For XY routing algorithm, due to the strange behaviours introduced in the earlier section, it is not taken into consideration in the comparison above.

Reference

DyAD – Smart Routing for Networks-on-Chip. Retrieved from:

https://ieeexplore-ieee-org.libproxy1.nus.edu.sg/stamp/stamp.jsp?tp=&arnumber=1322482

Evaluation ofPseudo Adaptive XY Routing Using an Object Oriented Model for NOC. Retrieved from:

https://ieeexplore-ieee-org.libproxy1.nus.edu.sg/stamp/stamp.jsp?tp=&arnumber=1590068

A Method to Improve Adaptivity of Odd-even Routing Algorithm in Mesh NoCs. Retrieved from:

https://ieeexplore-ieee-org.libproxy1.nus.edu.sg/stamp/stamp.jsp?tp=&arnumber=7445418

Comparison of NoC Routing Algorithms Using Formal Methods. Retrieved from:

https://pdfs.semanticscholar.org/7abe/070361b8bdcf0059fcce7a4eb8265e84f44e.pdf?_ga=2.206718928.290580094.1574064760-589498368.1574064760

Ma Junting	Wang Yawen
Brainstorm project topics	Brainstorm project topics
Analyse routing algorithms	Analyse routing algorithms
Implement routing algorithms – XY, West-First, North-Last, Neg-First, Odd-Even	Implement routing Algorithm - DyAD
Carry out simulation	Carry out simulation
Collect and compile simulation results	Work on final report
Work on final report	