University of Pisa and Scuola Superiore Sant'Anna

Performance Assessment of Schedulers for Optical Interconnection Networks

Master in Computer Science and Networking

Adviser: Prof. Isabella Cerutti

By: Mohammed Behredin Aman

April 29 2016

Abstract

University of Pisa and Scuola Superiore Sant'Anna

Master in Computer Science and Networking

By: Mohammed Behredin Aman

Performance Assessment of Schedulers for Optical Interconnection Networks

With ever-increasing demand for high-performance computing systems, interconnection networks, serving as the communication links in multicore architectures have become a key element for guaranteeing the system performance. Compared with bandwidth-limited power-hungry electrical interconnection networks, optical integrated interconnection networks also referred to as networks-on-chip ($ONoC$) architectures are emerging as a promising alternative to enable future computing performance.

In $ONoC$ architectures, scheduling algorithms are necessary for avoiding packet collisions while achieving high throughput, low latency, and good fairness. Scheduling algorithms exist for non-blocking electrical interconnection networks (NoC). These algorithms can be applied to $ONoC$, while accounting for additional constraints arising from optical component limitations.

In this thesis various scheduling algorithms are simulated, With the objective of comparing their latency and throughput using $C++$ programming language for $ONoC$ with bus and ring topologies.

An optimal scheduler based on two-step scheduling (TSS) technique is proposed. The optimal TSS models the scheduling problem in two steps for $ONoC$. The first step is the matching step which is done by representing each node pair as input bipartite graph then matching takes place between the input and output ports. The second step performs the wavelength assignment between each paired node while avoiding collisions and also with the consideration of wavelength continuity. The two-step approach with the $iSLIP$ and MWM algorithms are considered.

The proposed optimal TSS is simulated and its performances are evaluated. The optimal scheduler with maximum weighted matching (MWM) scheduling policy achieves better results in comparison to $iSLIP$ scheduling policy based on queue length under any packet arrival process. The optimal MWM scheduling policy achieved better performance for both bus and ring topologies.

The main result is that unidirectional ring topology outperforms the bus topology for any number of wavelengths less or equal to the number of $ONoC$ port, even if the average path length is longer. The reason is that in the bus topology half of the wavelengths are allocated in each direction, fixing the maximum number of packets in each direction using two transceivers per node can compensate this issue, reaching to better performance than the ring.

Acknowledgment

I owe my gratitude to all those people who have made this dissertation possible. It is because of them that I will cherish my experience of this work for the rest of my life.

My deepest gratitude is to my adviser Prof. Isabella Cerutti. I feel fortunate to have her as my adviser who gave me the freedom to explore on my own and at the same time the guidance to recover when my steps faltered. Prof. Isabella taught me how to question thoughts and express ideas. Her patience and support helped me sort out the technical details of my work. I am also thankful to her for encouraging the use of correct grammar and consistent notation in my writings and for carefully reading and commenting on countless revisions of this manuscript.

I am grateful to Prof. Marco Danelutto for his encouragement and for giving me permission to the university computers to run this work.

I gratefully acknowledge the extraordinary financial support by Scuola Superiore Sant'Anna. I am also grateful to Dr. Claudio Manfroni for his various forms of support during my graduate study.

I would like to acknowledge Associazione Sante Malatesta Onlus for giving me an accommodation.

Many friends have helped me stay sane through these difficult years. Their support and care helped me overcome setbacks and stay focused on my graduate study. I greatly value their friendship and I deeply appreciate their belief in me.

Most importantly, none of this would have been possible without the love and patience of my family. I would like to express my heart-felt gratitude to my beloved and understandable mother Medina Ahmedin and my sister Hayate Behredin, for supporting me in every aspect of my life.

Contents

List of Figures

1 Introduction

This chapter gives a general introduction on how energy efficient and energy proportional interconnection networks are needed in modern data centers and high performance computing systems. The challenges are introduced and solutions proposed in this thesis are detailed. The thesis organization is presented in the last section of this chapter.

1.1 Motivation of the Thesis

The performance of computing systems has continuously improved over the last years with increasing data processing and storage capabilities, as a result of the continuous technological improvement of the new generations microprocessors. Following the predictions of Moore's law, the continuous growth in the number of CPU transistors and the clock frequency boosted the evolution of computing systems [1].

To leverage the computational performance, explicit parallelism is exploited at the processor level as well as at the system level to realize high performance computing platforms. Computing platforms of different types offer tremendous computing and storage capabilities suitable for scientific and business applications.

A notable example is given by supercomputers, data centers enable fast retrieval of stored information for users connected to the Internet, and they can also support advanced applications (such as cloud computing) that offer computational and storage services. Other relevant examples are stated by the fastest computing platforms, used for running highly calculation-intensive applications in different scientific fields.

The increasing quest for information and computational capacity to support such applications is driving the performance growth, which is achieved by parallelism. The parallelism allows application tasks to be executed in parallel across multiple distinct processors leading to a reduction in the execution time and an increase in the computing platform utilization.

To benefit from such advantages, the computing systems should be interconnected through a high-capacity NoC. In currently deployed data centers and server farms, the parallelism is achieved by tightly clustering thousands of homogenous servers [2]. Typically the numerous racks hosting few tens of servers are connected through a rack switch usually placed at the top. The rack switches in turn connected to a cluster switch as shown in Figure 1 so that each server can communicate with any other server. The communication infrastructure consists of electrical NoC typically based on Ethernet (for lower cost and flexibility) or Infiniband protocol (for higher performance). Similarly in supercomputers, NoC with high throughput and low latency is required for connecting thousands of computing nodes [3].

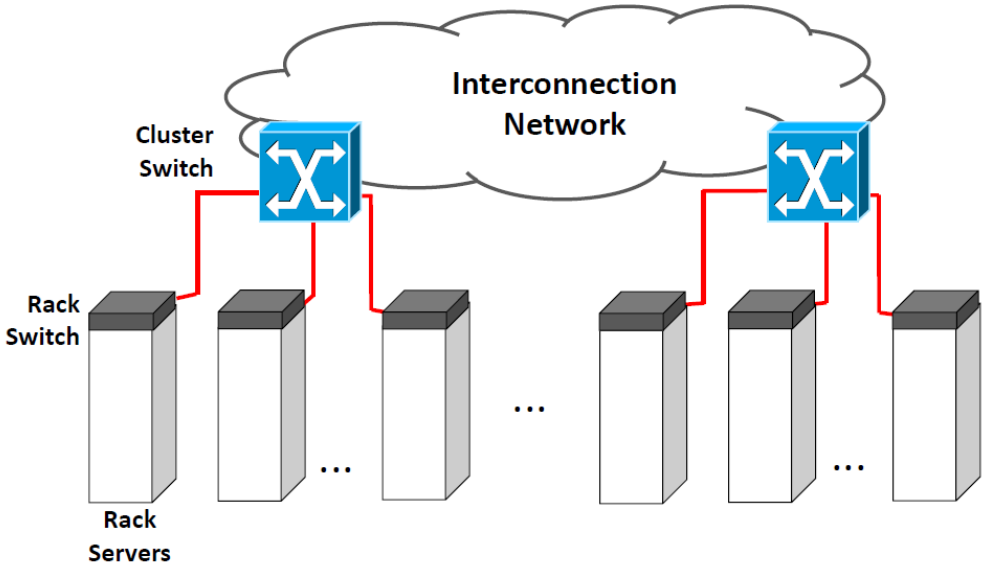

Figure 1: Generic interconnection network of a computing platform

Currently, high-performance scientific computation has been demonstrated in data centers by running tasks in parallel through cloud computing. But the performance of the communication infrastructure in such systems is lagging behind the expectations [4]. Furthermore, the performance requisites of high throughput and low latency are stringent especially for the high-performance computing tasks. In the last decade, the main bottleneck of computing infrastructure has shifted from the compute nodes to the performance of the communication infrastructure [5]. As computing platforms scale (e.g., with an increase in the number of servers and in the computational capacity) the requisites of high throughput and low latency are becoming more difficult to achieve and ensure.

Current *NoC* are based on electronics. Electronic *NoC* have several advantages i.e. they are cost-effective and can easily realized with high-volume integrated silicon-based devices. However, an increase in the processor speed, number of transistors on a chip and number of interconnected elements in a network are pushing the overall power consumption and dissipation of today's electronic *NoC* to their physical limit [6].

To overcome this bottleneck, *NoC* solutions based on optical technology have the potential to overcome the limitations of electronics by enabling high transmission rates with lower power consumption [3], [7], [8], [9]. Compared to electrical links, photonic point-to-point links enable much greater aggregated bandwidth-distance product, allowing increasing communication capacity. Although these solutions can help point-to-point transmissions, yet more complex architectures based on optics need to be researched on to further increase the performance. Indeed, the introduction of optics with in the *NoC* has been proposed by the scientific community and has been shown to achieve greater scalability and throughput compared to electronic switches [10]. Integration of the optical devices with electrical circuitry

is challenging but recent progress in the field of optical integration [11] indicates that the introduction of optics within NoC is expected to become a viable solution [12].

The design and realization of $ONoC$ remains however challenging due to the lack of effective solutions for all-optical buffering and processing. The power consumption of $ONoC$ can potentially be lower than electrical NoC but it is not yet negligible [13]. So while optical transmissions undoubtedly demonstrated the capability to handle tremendous amount of data traffic [14], it is now necessary to design scalable $ONoC$ architectures for data centers able to achieve both a high throughput at peak utilization and a low power consumption proportional to the NoC utilization levels [15].

1.2 Photonics solutions for Interconnection Networks

$ONoC$ provide connectivity between all shared elements of the computing platform (e.g. processors, storage elements) and allow switching in the optical domain. Shared elements can communicate with each other in different ways; either through distinct physical data paths (space switching) or using distinct wavelengths (wavelength switching) exploiting wavelength division multiplexing (WDM) or by allocating different time slots (time switching) to the packets destined to different output ports.

By exploiting a single domain for switching, single-plane architectures can be realized [16]. Optical wavelength-switched architectures are realized by exploiting the capability of the optical domain to accommodate multiple wavelengths in the electromagnetic spectrum through wavelength division multiplexing. Wavelength-switched architectures by taking the advantages of WDM technologies achieved transmitting packets on distinct wavelengths according to the desired destination port.

Usually single-plane switches can be built resorting to a single type of port or gating element, but the devices are characterized by some drawbacks as high crosstalk, high power loss, small bandwidth and high power consumption. The disadvantages ultimately limit the scalability of the switches. However, the combined use of more than one type of gating element can help scaling to switches with high port count [17].

1.3 Scheduling Issues

The $ONoC$ require a dynamic scheduler to decide which data packets to be switched in each wavelength.

However due to the limited bandwidth and the required spacing between the different wavelengths, the number of wavelengths can be smaller than the number of nodes, especially when scaling the $ONoC$ size. The $ONoC$ behaves as a blocking switching architecture; which means that one or more the routing requests to a free output port cannot be established without interfering with other traffic. The number of wavelengths might not be sufficient to

support a communication between each paired ingress and egress $ONoC$ ports. Therefore, accustomed scheduling frameworks devised for non-blocking switching architectures are not suitable for blocking architectures. As shown in Figure 2 the scheduling problem in $ONoC$ architecture can be viewed as bipartite graph matching problem, where input ports and output ports form two sets of disjoint nodes, and the requests form the edges. Each input port is equipped with Virtual Output Queue (VOQ) which is the technique used in input-queued switches where rather than keeping all traffic in a single queue, separate queues are maintained for each possible output location. It addresses a common problem known as head-of-line blocking.

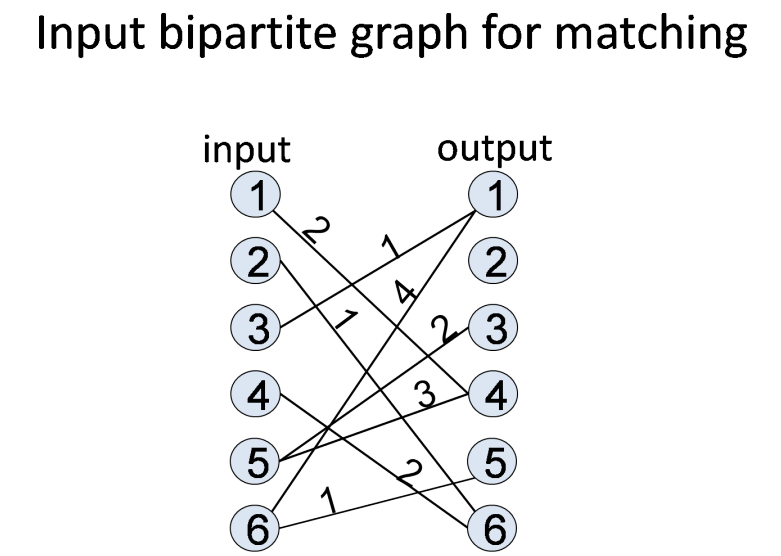

Figure 2: Input bipartite graph

In order to attain low-latency, high throughput and a fair access to the shared resources communications between paired ingress and egress ports must be scheduled properly. In this work it is assumed that data transmission from any egress port to any ingress port is based on synchronous fixed-sized packet switching: at each time slot a packet can be transmitted and switched between each port pair.

When considering an $ONoC$, the propagation time of a packet can be considered negligible with respect to the transmission time. This means that in a single time slot a packet or a flit can be transmitted along multiple links and thus the selected wavelength must be reserved along the path for the whole duration of the time slot. The scheduling problem aims to select the ingress or egress port matching, and the wavelength assignment on which the matched ports able to transmit packets or flits. For generalization the scheduling problem consists two sub-problems: The matching problem and the wavelength assignment problem:

- The matching problem aims to select a VOQ for each input port, that is the ingress

and egress port pair. As a consequence at most one packet is transmitted from each port or received at each port.

- The wavelength assignment problem goal is to assign wavelengths to the matched transmission form, as a consequence wavelength coherence is granted and contiguous reuse is exploited without collisions.

1.4 State of the Art

The scheduling problem has been well studied in the past for synchronous non-blocking switches. Most of the proposed approaches are based on the utilization of the $VOQs$ at each input port, avoiding the head-of-line blocking issue. The proposed scheduling frameworks can be divided into three main classes aiming at finding an optimal, maximal, and randomized matching.

Optimal approaches aim at optimally solving the matching problem, more specifically the maximum weighted matching (MWM). Link weights (e.g., VOQ size or longer waiting time in VOQ) [19] or node weights (e.g., a combination of VOQ size at a node) [20] can either be used as weights. It has been proved that the MWM algorithm achieves 100% throughput under Bernoulli i.i.d. packet arrival process, uniform or non-uniform [18], [23]. Later, the results were also extended for more general arrival processes and admissible traffic [28]. Additionally the algorithm also provides low delay. However, their good performance and stability come at the expense of high computation complexity ($O(N^3)$) [19] which can be reduced to $O(N^{2.5})$ when using the algorithm proposed in [20]. Such high complexity motivated the search for faster scheduling approaches.

Different from maximum matching, maximal matching algorithms aim at approximating maximum size matching through iterative, fast, and simple to implement algorithms. Maximal matching algorithms can provide 100% throughput under uniform traffic and fairly good delay performance as well. However, they are not stable under non-uniform traffic. Notable examples are iSLIP algorithm [22], ϵ-auction and ϵ-min-sum algorithms [26], frame-based maximal weight matching [27], synchronous round robin [25].

Randomized matching algorithms [29] have been designed with objectives of stability and to approximate MWM. Randomized algorithms are linear in complexity and provide the benefit of being stable under any admissible traffic as well. However, the delay encountered is higher than that of approximating algorithms, as randomized algorithms have been designed with objectives of stability rather than small average delay. All the above mentioned approaches are suitable for NoC that are non-blocking or re-arrangeably non-blocking. However for the NoC architectures realized with integrated photonics additional constraints may arise in the matching problem, leading to internally blocking performance [24].

Thus, whereas for electronic NoC the scheduling problem is well defined and the theoretical methodology (e.g., stability analyses) is well established and numerous approaches were de-

veloped and assessed, for $ONoC$ with internal blocking the scheduling problem is intimately related to the architecture [21]. Moreover, an extension or evolution of the existing theoretical methodology and approaches is required, to make it suitable for supporting multiple domains and the internal blocking.

1.5 Thesis Organization

This thesis is organized as follows.

Chapter 2 presents the $ONoC$ architecture. Two alternative implementations are considered differing in the type of optical devices used to acquire the wavelengths. i.e. either: 1)Fixed or 2)Tunable. By using these futures of photonic integrated architectures, implemented two different topologies, which are unidirectional ring and unidirectional buses. The unidirectional buses are implemented using two different designs one transmitter or two transmitters per node.

Chapter 3 presents the TSS framework implemented for the $ONoC$. Two alternative implementations are considered which performs the scheduling in two steps i.e. 1) the first step is the matching step which is done by representing each node pair as input bipartite graph, then matching takes place between the input and output ports 2) the second step performs the wavelength assignment between each paired node while avoiding collisions and also with the consideration of wavelength continuity. The two-step approach with the $iSLIP$ and MWM algorithms are considered.

Chapter 4 presents the experimental results for the two-step scheduling framework based on the considered topologies. The simulator collects the statistics in steady state condition. The performance metrics considered are latency and the throughput.

Chapter 5 presents the conclusion of the thesis based on the experimental results collected and discusses the future work.

2 Photonics Integrated Networks on Chip

This chapter is devoted to discuss the $ONoC$ architectures. Based on this architecture two alternative implementations are considered differing in the type of optical devices used to acquire the wavelengths which are either: 1) Fixed or 2) Tunable. Two different topologies are considered which are unidirectional ring and bus. The unidirectional bus are implemented using two different designs one transmitter or two transmitters per node.

2.1 ONoC Implementations

$ONoC$ are required to offer high performance in terms of latency and bandwidth while keeping the footprint and power consumption limited.

The silicon-based Photonic Integrated Circuit (PIC) realization of the $ONoC$ enables multiple transmissions on the same wavelength with low crosstalk. Parallel transmissions of packets (flits) on the same wavelength when their paths span is disjoint or on different wavelengths. In wavelength switched architectures the number of wavelength channels that can be used is limited and depends on the wavelength spacing and optical bandwidth of the photonic devices.

As shown Figure 4 below each input or ingress port is equipped with a transmitter Ti (i.e. laser and modulator) that converts the electronically stored data into an optical signal. Each output or egress port is equipped with a broadband photo receiver Ri that allows to receives an optical signal on any wavelength of the operating band. Each transmitter (Ti) or Receiver (Ri) is connected to a local microring, that enables the filtering and adding or dropping of the signal transmitted in the network topology.

Two types of $ONoC$ implementations are envisioned:

- **Fixed transmitters**:

  The Fixed transmitter implementation operates on a fixed wavelength which does not allow the functionality of adjusting or tuning the wavelengths in the case of collisions or transmissions. This architecture requires the wavelength allocation between each paired nodes when the number of nodes equal to the number of wavelengths, but in case the number of wavelengths are less than the number of nodes there could be a blocking between paired nodes.

- **Tunable transmitters**:

  The Tunable transmitter implementation operates on the whole band or a set of wavelengths which is equivalent with a set of fixed wavelengths, each one operating on a distinct wavelength of the band. This architecture allows the functionality of adjusting or tuning the wavelengths.

2.2 Ring topology

Figure 3 shows the unidirectional ring topology with one transmitter per node. The ring connects the ports and supports W wavelengths in one direction only. The ring topology is indeed realized with a larger central microring resonator. Microring resonators are also used at the local ports for adding (dropping) the optical signal from (to) the port to (from) the shared ring. Add and drop operations are achieved by properly tuning the local microring resonator to the wavelength to be added (dropped). The multi wavelength communication is also possible on the ring with low crosstalk by properly aligning the resonant frequencies of the central microring and of the local microring resonators, enabling beneficial filtering effects [30], [31].

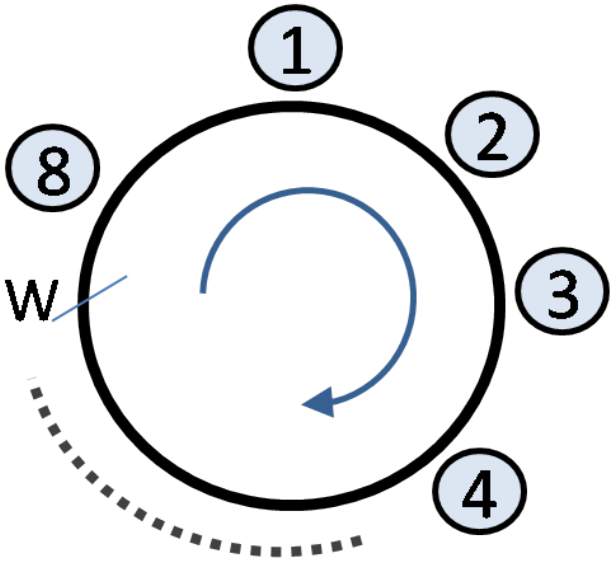

Figure 3: Unidirectional ring with one transmitter

In the transmission of a packet (flit) the continuity of the selected wavelength should be respected along the path between the ingress and egress port through the considered (in this case Ring) topology. As an example in Figure 4 the transmissions shows on the same wavelength on different links and on different wavelengths on the same link. Transmissions between $T1 - R2$ and $T2 - R4$ happen on the same wavelength while transmissions $T2 - R4$ and $T3 - R1$ use different wavelengths and they can both pass on the same link from $T3$ to $R4$.

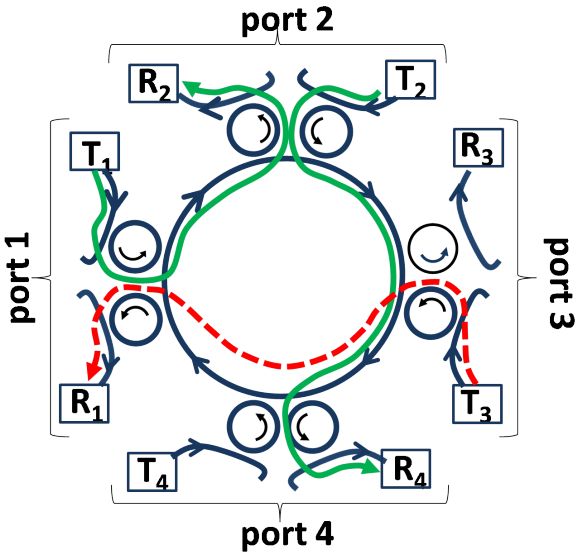

Figure 4: Input or Output port in the $ONoC$

2.3 Bus topology

Figures 5 and 6 show the unidirectional bus topology connecting ports and supporting half of the wavelengths in one direction and the other half wavelengths in the other direction. Two configurations are possible:

- **Buses with one transmitter per node**

  The unidirectional bus topology is indeed realized with two parallel waveguides used in opposite directions. As shown in Figure 5 it is assumed that each waveguide carries $W/2$ wavelengths (W even). $ONoC$ architectures are used at the local ports for adding (dropping) the optical signal from (to) the port to (from) the incoming (outgoing) link. Add and drop operations are achieved by properly tuning the $ONoC$ architectures to the wavelength to be added (dropped). The multi wavelength communication is also possible on the bus with low crosstalk by properly aligning the resonant frequencies [30], [31].

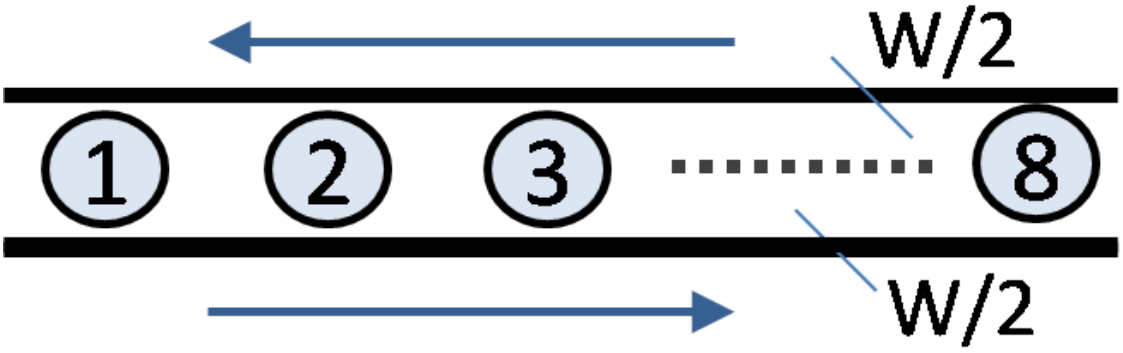

Figure 5: Unidirectional bus with one transmitter

In the transmission of a packet or flit the continuity of the selected wavelength should be respected along the path between the ingress and egress port through the bus topology.

- **Buses with two transmitters per node**

  A single transmitter and a single receiver are the minimal requirements for a single-hop system but the protocol and the systems performance can be improved by equipping nodes with multiple transceivers.

  Figure 6 shows a unidirectional bus topology realized with two parallel waveguides operating in opposite directions and with two transmitters per node. The bus connects the ports and supports $(W/2)$ half of the wavelengths in one direction and half of the wavelengths in the other direction. At each node one transceiver operates on a waveguide and the other transceiver on the opposite waveguide.

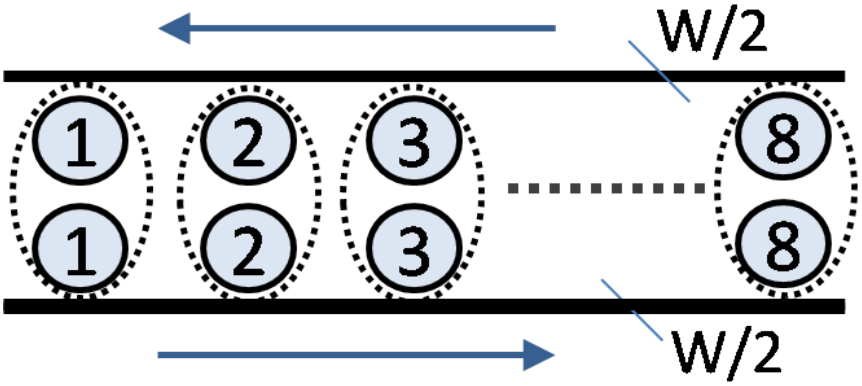

Figure 6: Unidirectional bus with two transmitters

3 Two-step scheduling framework

This chapter presents the two-step scheduling framework for the $ONoC$. Two alternative implementations are considered which performs the scheduling in two steps; 1) the first step is the matching step which is done by representing each node pair as input bipartite graph and by matching the input and output ports 2) the second step performs the wavelength assignment between each node pair while avoiding collisions and also with the consideration of wavelength continuity. The two-step approach with the $iSLIP$ and MWM algorithms is considered for the first step.

3.1 First step: Matching

Figure 7 shows the matching step that consists in finding the match of input ports with output ports having the highest weight [32]. Thus, it can be defined as a *maximum weighted matching* on a bipartite graph. In the bipartite graph shown in Figure 2 the nodes represent the ports. A link (i, j) is added between input port i and output port j, if input port i has at least one packet for output port j, stored in the corresponding VOQ. The edges connecting the nodes of a bipartite graph have weights associated to queue lengths or other metrics [6], [19] according to the scheduling policy adopted (e.g., fairness, delay reduction, bounded delay).

The matching problem can be addressed with different well-known scheduling algorithms that trade optimality for computational complexity. Two of them are used for this research and described below.

3.1.1 MWM Algorithm

A MWM algorithm finds the matching at highest weight. This algorithm can give preference to queues with a larger occupancy or to packets that have been waiting longest. It depends on the weight of the links in the bipartite graph. For this thesis MWM algorithm using the queue length is considered.

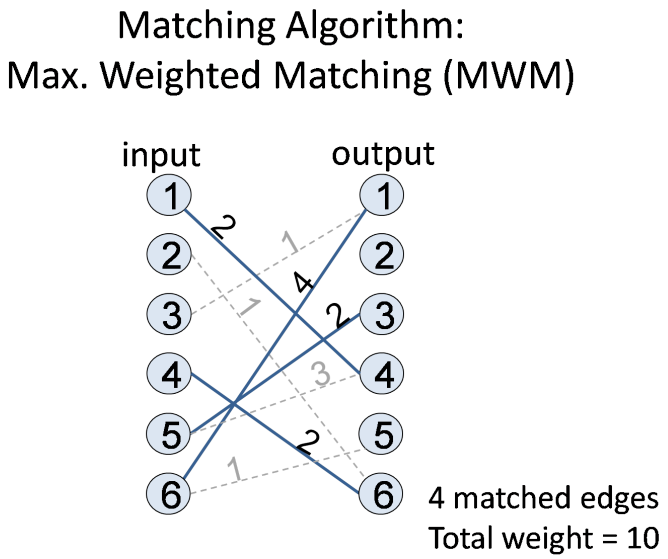

Figure 7: The matched edges for MWM algorithm

It has been proven that the MWM algorithm achieves 100% throughput under any packet arrival process [19], [23], [28]. Additionally the algorithm also provides low delay. However, its good performance and stability come at the expense of high computational complexity $O(N^3)$ [19] which makes this scheduling algorithm prohibitively expensive for practical implementation in high speed switches.

3.1.2 iSLIP Algorithm

Different from maximum matching, maximal matching algorithms aim at approximating maximum size matching through iterative, fast and simple to implement algorithms. They can provide 100% throughput under uniform traffic and fairly good delay performance as well. However, they are not stable under non-uniform traffic. PIM (Parallel iterative matching) and $iSLIP$ algorithms belong to this category.

For this thesis the $SLIP$ algorithm with multiple iterations is considered and it is called $iSLIP$ (iterative $SLIP$). As an example the $iSLIP$ solution for the first iteration is shown in Figure 10. Each iteration attempts to add matches not made by earlier iterations as shown in Figure 11. Matches made in one iteration are never removed by a later iteration even if a larger sized match would result as shown in Figure 12. The three steps of each iteration operate in parallel on each output and input described as follows:

- Step1. **Request**: Each unmatched input sends a request to every output for which it has a queued packet
- Step2. **Grant**: If an unmatched output receives any request it chooses the one that

appears next in a fixed round-robin schedule starting from the highest priority element. The grant step of $iSLIP$ is shown in Figure 8. The output notifies each input whether or not its request was granted. Pointer gi pointing to the highest priority element of the round-robin schedule. The pointer gi is incremented ($moduloN$) to one location beyond the granted input if and only if the grant is accepted in Step3 of the first iteration.

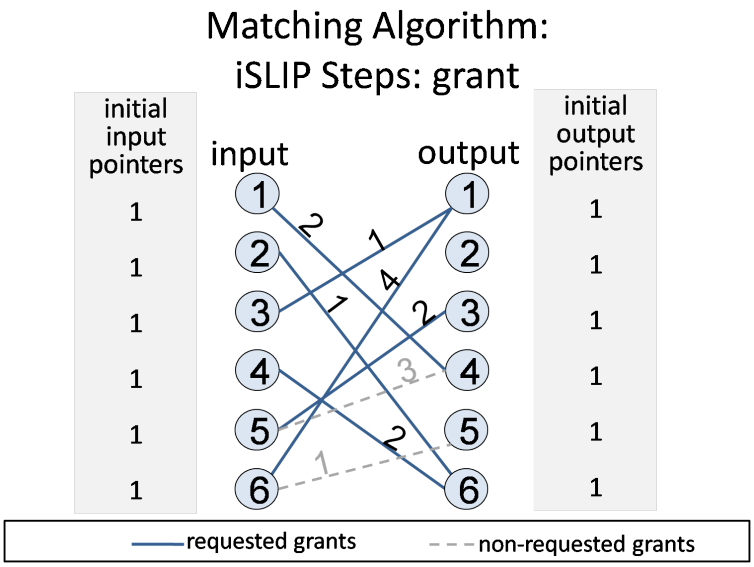

Figure 8: iSLIP Algorithm grant step

- Step3. **Accept**: If an unmatched input receives a grant it accepts the one that appears next in a fixed round-robin schedule starting from the highest priority element. The accept step of $iSLIP$ is shown in Figure 9.

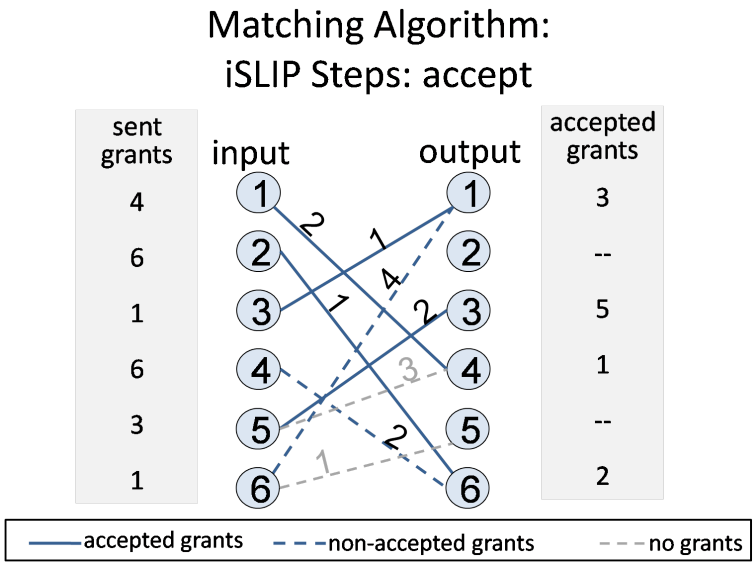

Figure 9: iSLIP Algorithm accept step

The pointer ai to the highest priority element of the round-robin schedule is incremented (modulo N) to one location beyond the accepted output only if this input was matched in the first iteration.

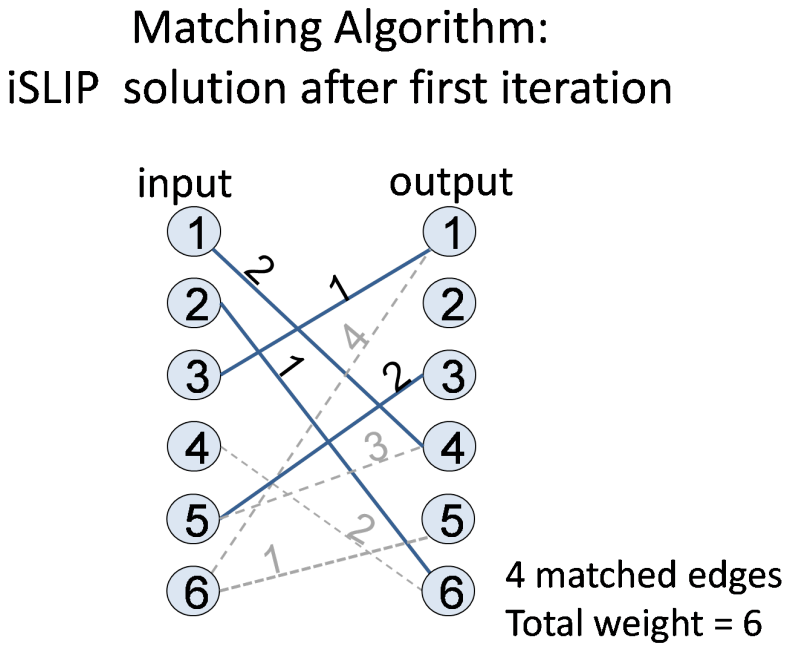

Figure 10: iSLIP Algorithm after first iteration

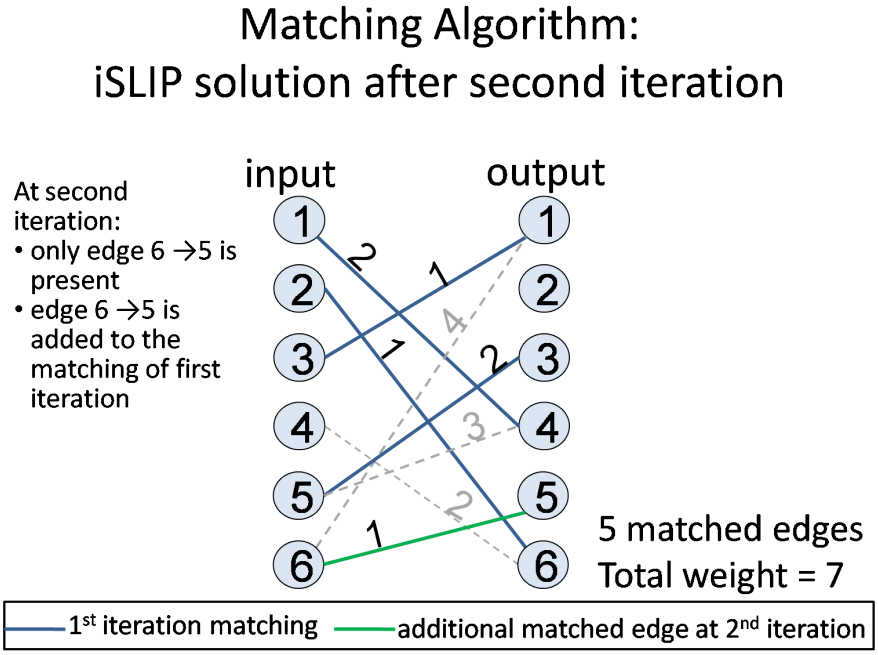

Figure 11: iSLIP Algorithm after second iteration

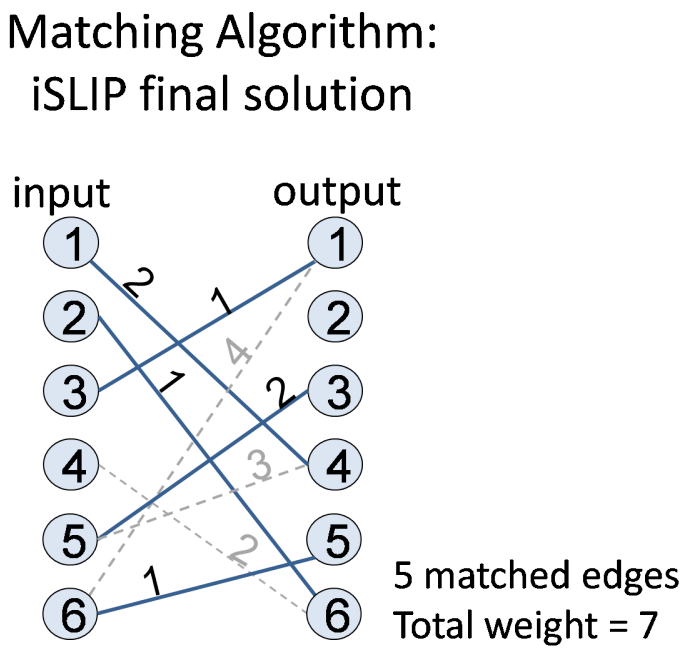

Figure 12: iSLIP Algorithm final solution

3.2 Second step: Wavelength Assignment

For this step the first-fit wavelength assignment algorithm is implemented. The wavelength assignment step consists in selecting a wavelength for each path of the matched port-pairs and in removing the transmissions that would lead to collision. Based on the type of the transmitters (tunable or fixed), two types of wavelength assignment techniques can be applied, which is fixed and tunable wavelength assignment. Fixed wavelength assignment implementation applied for the fixed $ONoC$ implementation. Where as tunable wavelength assignment is applied for tunable $ONoC$ implementation.

Fixed wavelength assignment allows the wavelength allocation between each node pairs when the number of nodes equals the number of wavelengths, but in case of lower number of wavelengths there could be a blocking as shown in Figure 13.

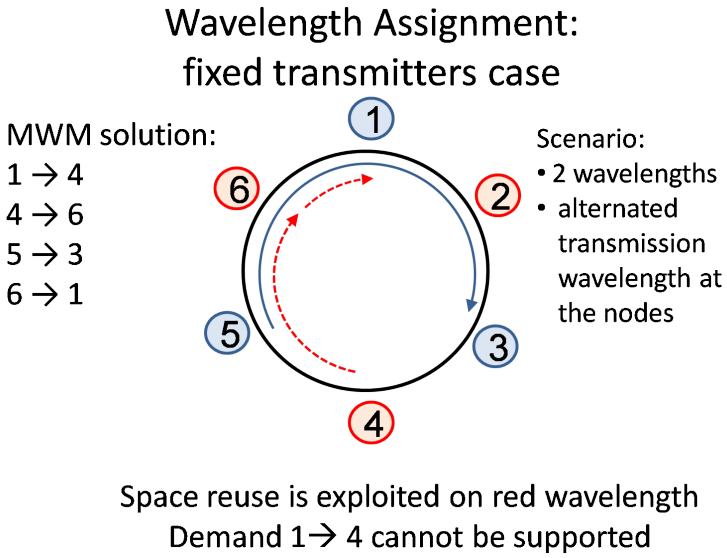

Figure 13: Wavelength assignment for fixed unidirectional ring

Tunable wavelength assignment allows the wavelength allocation between each node pairs when the number of nodes equals the number of wavelengths. As shown in Figure 14, when the number of wavelength is less than the number of nodes there could be blocking. To solve this issue the longest path first policy is applied and then the paths are assigned the wavelengths in a first-fit manner. Thanks to the tunability features of this implementation, it allows to use (tune) any available wavelengths.

Wavelength assignment on tunable implementations can be divided in two different scenarios, depending on the number of wavelengths to the ports ratio:

- **Wavelength / Number of ports equals to one**: in this situation, since each port can receive and transmit at most one packet per time slot, it is possible to assign

a single wavelength to each ingress or egress port without loosing of flexibility and performance.

- **Wavelength / Number of ports less than one**: in this situation, the matching and wavelength assignment problem need to be solved concurrently. In the wavelength assignment problem one of the wavelength is selected from the set of wavelengths for the paired ingress and egress port pair using a first-fit wavelength assignment techniques, while avoiding packet or flit collisions.

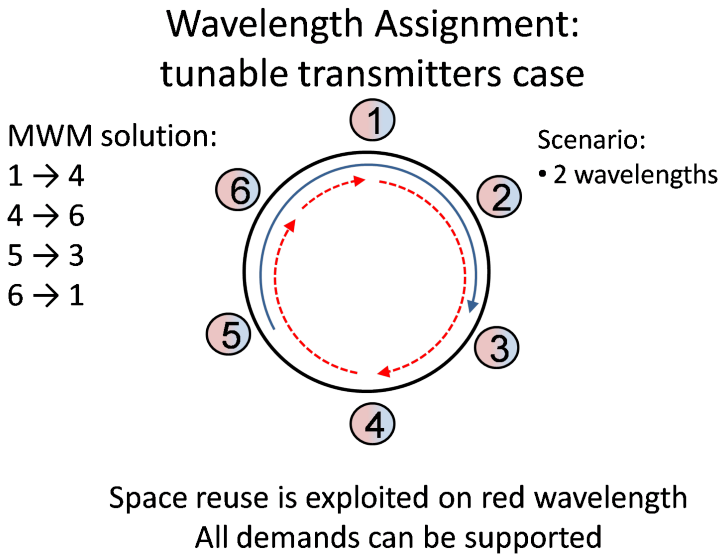

Figure 14: Wavelength assignment for tunable unidirectional ring

If no wavelength is available on all the links of the path, the matched port-pair is removed from the list, i.e. the corresponding packet cannot be transmitted in the considered time slot, but will be considered in the subsequent time slot. In general first-fit wavelength assignment requires global knowledge about the network. A database is needed to keep the information about resource reservation. Thus a centralized scheduler is envisioned. The computational complexity of the first-fit wavelength assignment is $O(N^2W)$ [33].

4 Experimental Results

The performance of the optimal TSS framework is assessed using a custom-built event-driven simulator implemented in C/C++. The average queuing latency (i.e., average number of time slots spent by the packet in the buffer before its transmission) is collected for a unidirectional bus and ring topologies using various configurations of wavelengths (i.e., $W = 4$ or 8) based on number of nodes equals to eight(i.e., N =8). Time slot duration is normalized to 1 which corresponds to the transmission time of one packet plus guard time. For fixed transmitter case, the input ports are allocated the different wavelengths in a round-robin fashion. Packets are generated at each port according to a Bernoulli process. For $iSLIP$ algorithm, four iterations are performed.

The simulations are run for large number of packets until the confidence interval of the average delay is less than 5% for confidence level of 95%.

4.1 MWM Algorithm

Figure 17 and 18 shows the latency comparison as a function of the $ONoC$ load when MWM algorithm is used in the first step. The comparison of the average queuing delay of the packets against different traffic loads. Three types of topologies are considered which are unidirectional buses with one transmitter per node Bus_1, unidirectional buses with two transmitters per node Bus_2 and unidirectional ring with one transmitter per node $Ring$. The fixed and tunable transmitters alternatively considered in each topology.

Figure 18 shows that fixed Bus_1 has higher latency than $Ring$. Figure 15 though shows the average hop length for Bus_1 i.e. (N+1)/3 versus Figure 16 shows N/2 for $Ring$. This unexpected result is due to the fact that in Bus_1 only half of the wavelengths are available for packets in one transmission directions and thus unbalanced scheduling of more W/2 packets in one direction may not be supported.

On the other hand, the higher flexibility of $Ring$ in supporting also unbalanced scheduling leads to a higher throughout and lower latency with respect to Bus_1. Alternatively, the use of two transceivers per node as in Bus_2 can compensate the limitations of Bus_1. Bus_2 outperforms $Ring$ at the expenses of a higher cost, larger footprint and higher power consumption. As shown on Figure 17 the reduction of throughput can be in part compensated by using tunable transmitters. Indeed tunability is more advantageous when the number of wavelengths is $W < N$.

The average hop length for unidirectional bus

$$\sum_{i=1}^{N-1}\sum_{j=i+1}^{N}(j-i)\cdot\frac{2}{N(N-1)}=$$
$$\frac{2}{N(N-1)}\left(\sum_{i=1}^{N-1}\sum_{j=i+1}^{N}j-\sum_{i=1}^{N-1}\sum_{j=i+1}^{N}i\right)=$$
$$\frac{2}{N(N-1)}(\sum_{i=1}^{N-1}(\sum_{j=1}^{N}j-\sum_{j=1}^{i}j-\sum_{j=1}^{N}i+\sum_{j=1}^{i}i))=$$
$$\frac{2}{N(N-1)}\sum_{i=1}^{N-1}\left(\frac{N(N+1)}{2}-\frac{i(i+1)}{2}-i\cdot N+i^2\right)=$$
$$\frac{2}{N(N-1)}\left(\frac{N^{N-1}(N+1)}{2}-\sum_{i=1}^{N-1}i-\left(N+\frac{1}{2}\right)+\sum_{i=1}^{N-1}\frac{i^2}{2}\right)=$$
$$(N+1)-\frac{\left(N+\frac{1}{2}\right)(N-1)}{(N-1)}+\frac{\left(N-1\right)(2N-1)}{6(N-1)}=$$
$$(N-1)\frac{\left(6N-6N-3+2N-1\right)}{6(N-1)}=$$
$$(N-1)\frac{\left(2N+2\right)}{6(N-1)}=$$
$$\frac{\left(N+1\right)}{3}$$

Figure 15: Unidirectional bus average hope length

The average hop length for unidirectional ring

$$\sum_{i=1}^{N}\sum_{i=1}^{N-1}j\cdot\frac{1}{N(N-1)}=$$
$$\sum_{i=1}^{N}\frac{N(N-1)}{2}\cdot\frac{1}{N(N-1)}=$$
$$\frac{N}{2}$$

Figure 16: Unidirectional ring average hope length

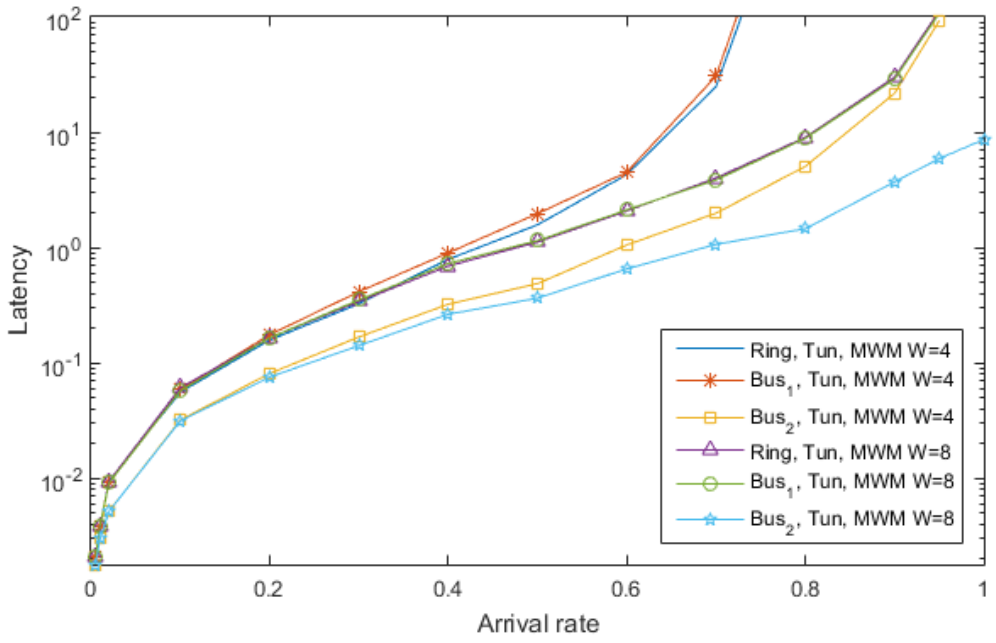

Figure 17: Tunable transmitters: latency vs. load for W=4, 8 and first step MWM

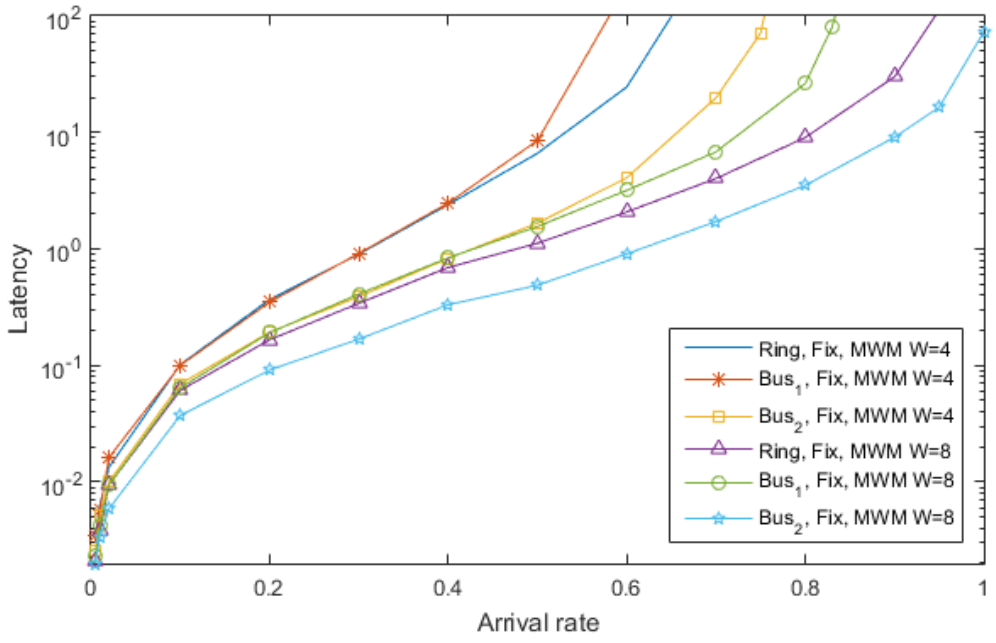

Figure 18: Fixed transmitters: latency vs. load for W=4 and 8 and first step MWM

Figure 19 shows the performance for MWM with fixed and tunable architecture with number of wavelength equals to four. The tunable Bus_1 has the same latency as the fixed Bus_1 at low traffic load. When the traffic load increases the latency of the fixed Bus_1 becomes higher than the tunable Bus_1. The tunable Bus_2 has better latency with respect to the other topologies for all traffic loads. When the traffic load increases the latency of the fixed Bus_2 becomes better than the others.

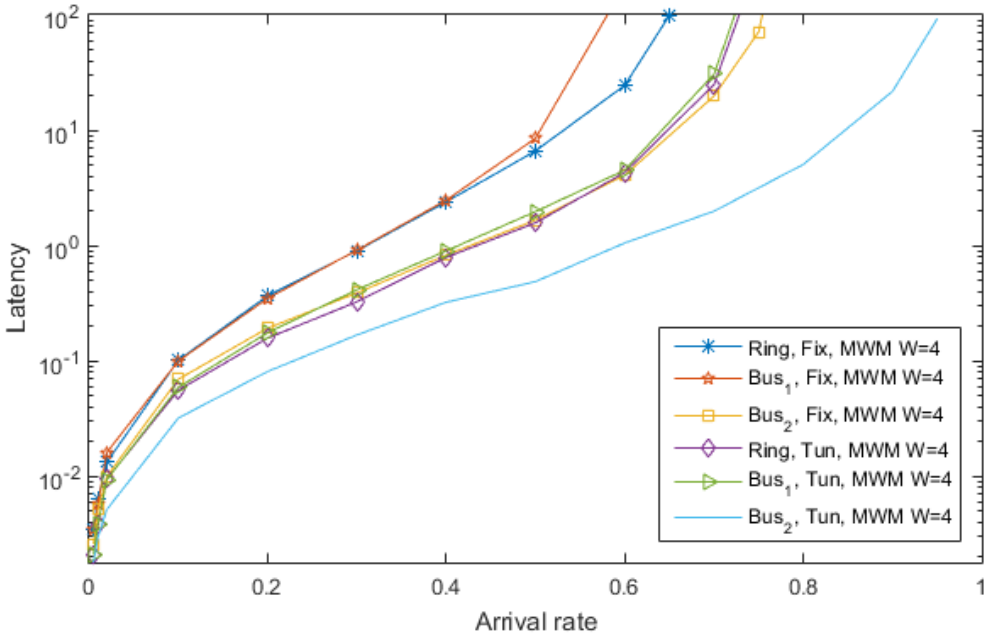

Figure 19: Latency vs. load for $W = 4$ and first step MWM

Figure 20 shows the performance for MWM with fixed and tunable architecture with number of wavelength equals to eight. When the number of wavelength equals to the number of nodes tunability is not required and thus the tunable transmitter design performs as fixed design. Figure 20 also shows the advantage of tunable architecture i.e. tunable Bus_1 with MWM performs equal with the tunable $Ring$ with MWM .

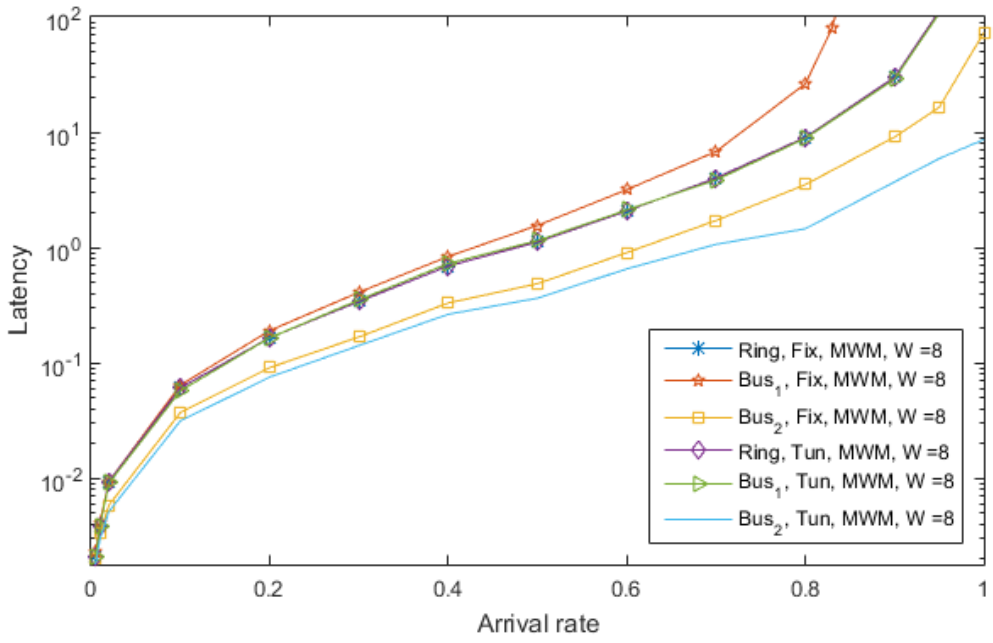

Figure 20: Latency vs. load for $W = 8$ and first step MWM

4.2 iSLIP Algorithm

The MWM algorithm performance and stability come at the expense of high computational complexity. Such high complexity motivated to brought the reference algorithm $iSLIP$.

Figure 21 and 22 shows the latency comparison as a function of the $ONoC$ load when $iSLIP$ algorithm is used in the first step.

Figure 21 shows the tunable architecture with number of wavelength equals to four and eight. The tunable $Ring$ with $iSLIP$ performs better than tunable Bus_1 with $iSLIP$. The computationally efficient $iSLIP$ algorithm shows a slight performance degradation when the traffic load increases.

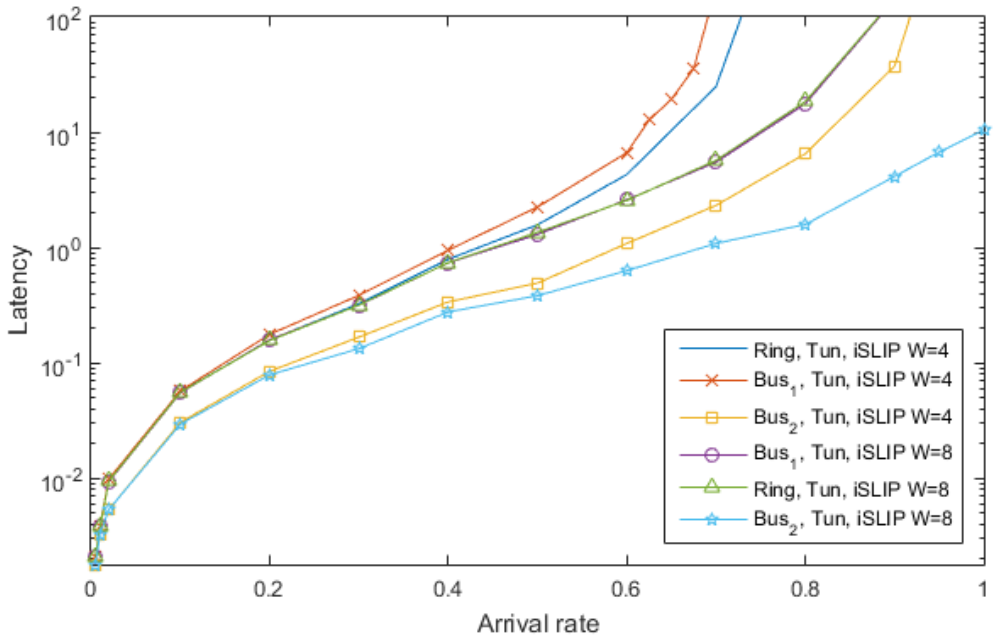

Figure 21: Tunable transmitters: latency vs. load for $W = 4$, 8 and first step $iSLIP$

Figure 22 shows the fixed architecture with number of wavelength equals to four and eight. The fixed $Ring$ with $iSLIP$ performs better than fixed Bus_1 with $iSLIP$ when the traffic load increases. Figure 22 also shows the advantage of having more number of wavelength i.e, $Ring$ or Bus_1 with wavelength equals to eight performs better than $Ring$ or Bus_1 with wavelength equals to four.

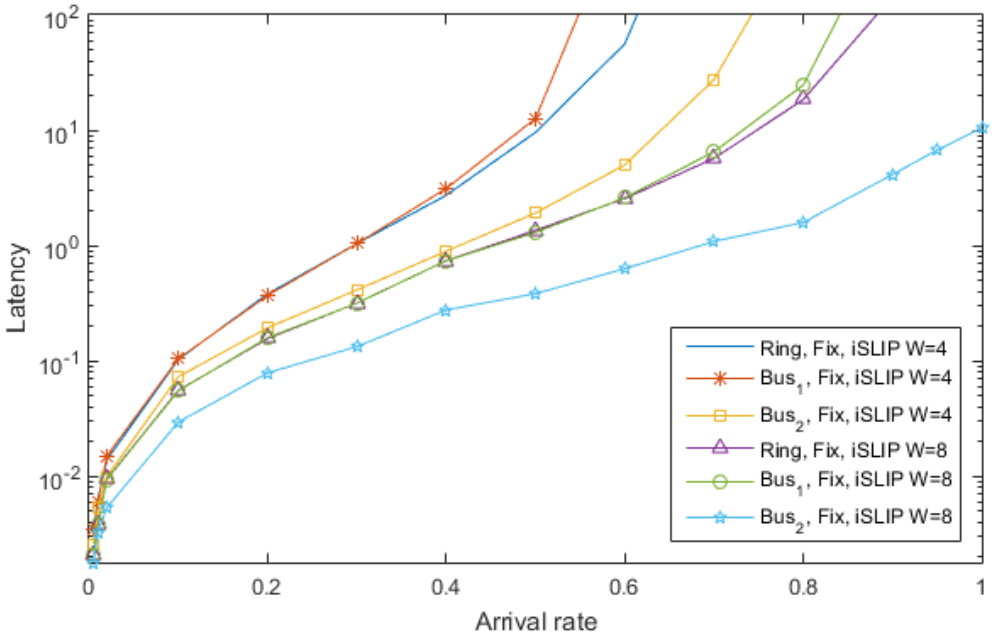

Figure 22: Fixed transmitters: latency vs. load for $W = 4$, 8 and first step $iSLIP$

Figure 23 shows the comparison for fixed and tunable $Bus_1$with wavelength equals to four. The tunable Bus_1 has the same latency as the fixed Bus_1 at low traffic load. When the traffic load increases the latency of the fixed Bus_1 becomes higher than the tunable Bus_1.

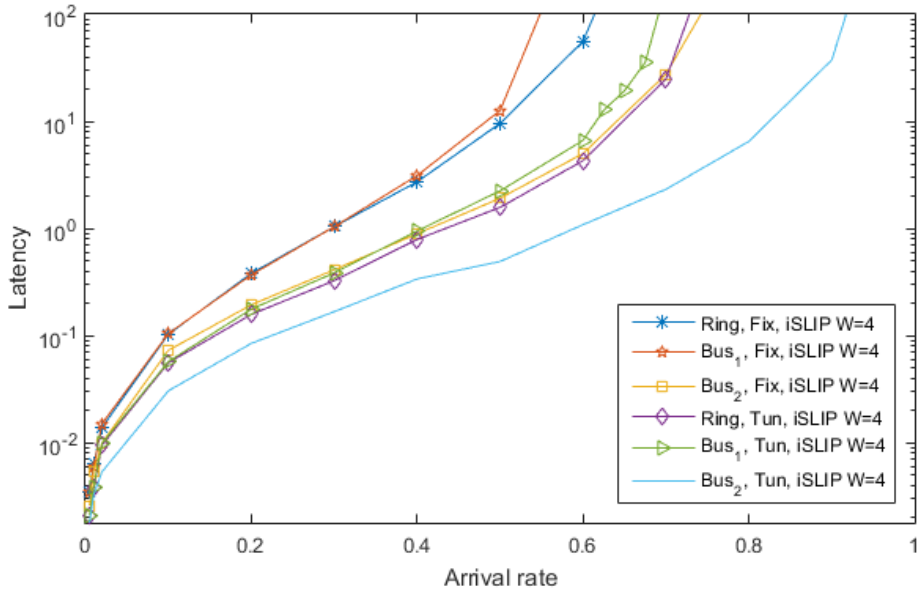

Figure 23: Latency vs. load for $W = 4$ and first step $iSLIP$

Figure 24 shows the comparison for *Ring* topology using tunable and fixed architecture. When the number of wavelength equals the number of nodes, tunability is not required and thus the tunable transmitter design performs as fixed transmitter. Figure 24 also shows the advantage of tunable architectures i.e. tunable Bus_1 with $iSLIP$ performs equal as *Ring* with $iSLIP$.

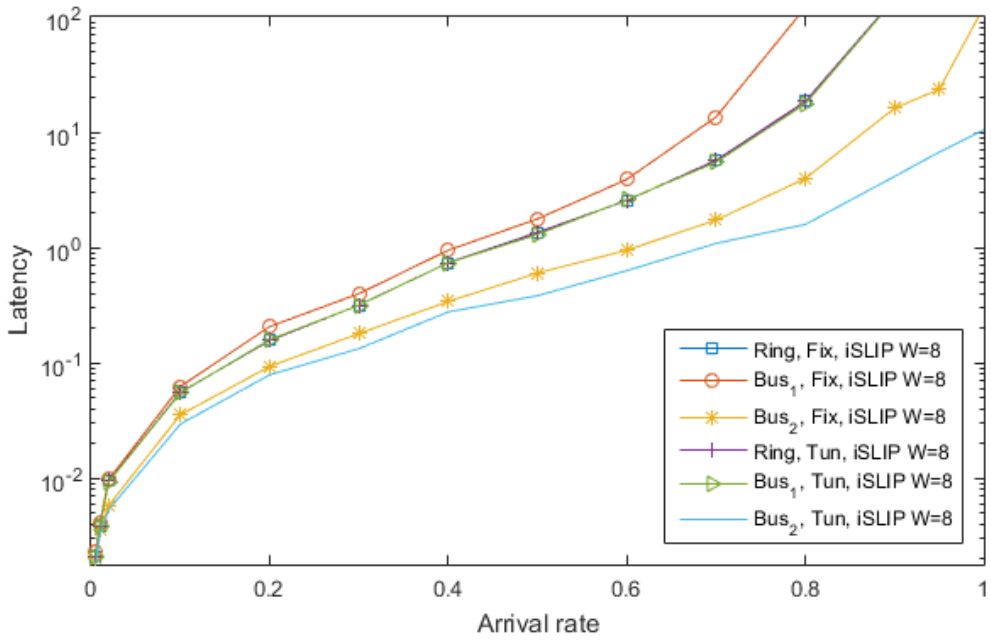

Figure 24: Latency vs. load for $W = 8$ and first step $iSLIP$

4.3 Comparison between MWM and iSLIP Algorithms

Figure 25 and 26 shows the latency comparison as a function of the $ONoC$ load when MWM and $iSLIP$ algorithms are used in the first step.

Figure 25 shows the performance of tunable design with MWM and $iSLIP$ when the number of wavelength equals to eight. The MWM performance is equal as the $iSLIP$ algorithm at low traffic loads. When the traffic load increases the MWM algorithm performs better than the $iSLIP$ algorithm, even the Bus_1 with MWM performs better than the *Ring* with $iSLIP$. *Ring* with MWM and Bus_1 with MWM performance are equal for all traffic loads. *Ring* with $iSLIP$ and Bus_1 with $iSLIP$ performance are also equal for all traffic load. Bus_2 with $iSLIP$ and MWM performance are almost equal just a slight degradation by $iSLIP$ after the traffic load of 0.95.

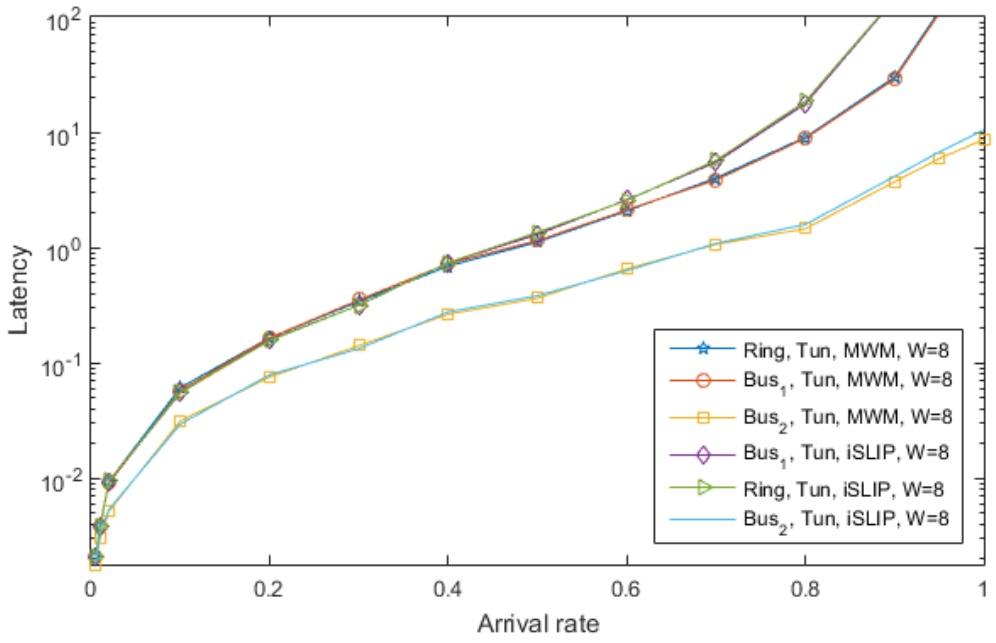

Figure 25: Tunable transmitters: latency vs. load for $W = 8$ and first step $iSLIP$ or MWM

Figure 26 and 27 show the performance of MWM and $iSLIP$ in the $ONoC$ when varying the number of wavelengths.

Figure 26 shows the performance of fixed design with MWM and $iSLIP$ when the number of wavelength equals to eight. At low traffic loads all topologies with one transmitter have equal performance. When the traffic loads are increase Bus_1 with $iSLIP$ shows a slight degradation with respect to $Ring$ with $iSLIP$, and also with respect to $Ring$ and Bus_1 of MWM.

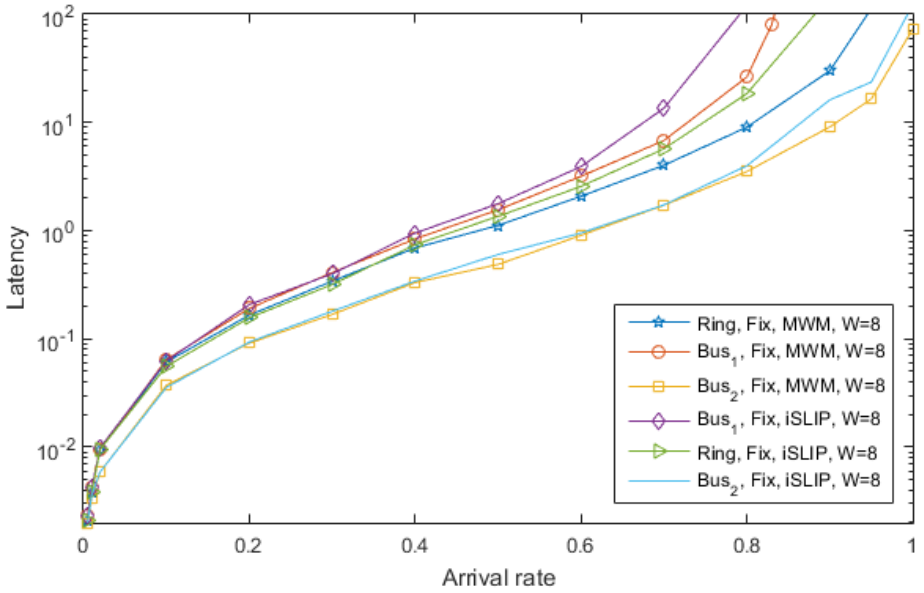

Figure 26: Fixed transmitters: latency vs. load for $W = 8$ and first step $iSLIP$ or MWM

Figure 27 shows the performance of fixed design with MWM and $iSLIP$ when the number of wavelength equals to four. At low traffic loads all topologies with one transmitter have equal performance.

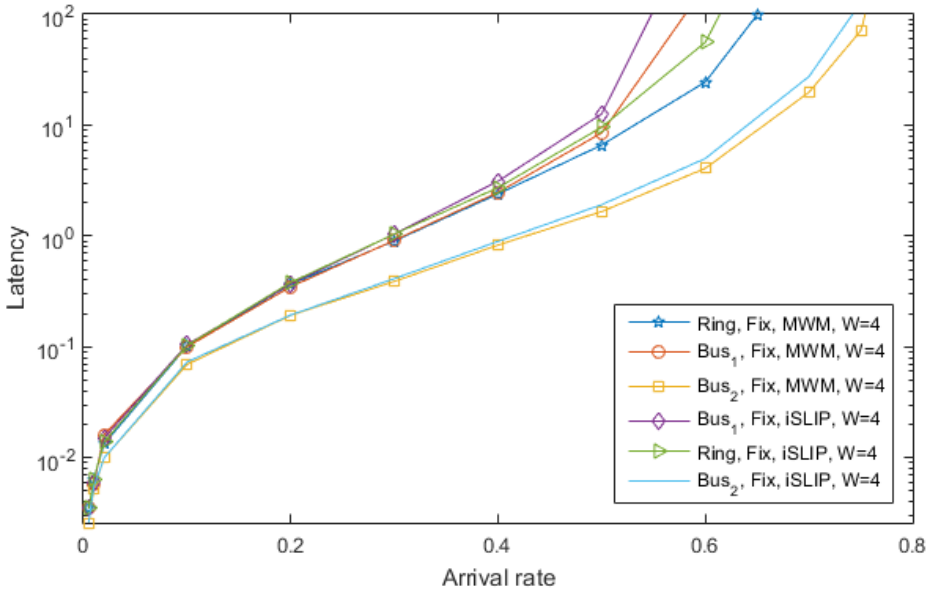

Figure 27: Fixed transmitters: latency vs. load for $W = 4$ and first step $iSLIP$ or MWM

Figure 28 shows the performance of tunable design with MWM and $iSLIP$ when the number of wavelength equals to four. Bus_1 and $Ring$ with $iSLIP$ are performing almost equal for all traffic loads. When traffic load is greater than 0.5 the Bus_1 with $iSLIP$ starts to show a slight degradation.

Bus_2 with MWM performance also the same at low traffic load with the Bus_2 with $iSLIP$. When the traffic load is greater than 0.7 the $iSLIP$ start to show some degradation.

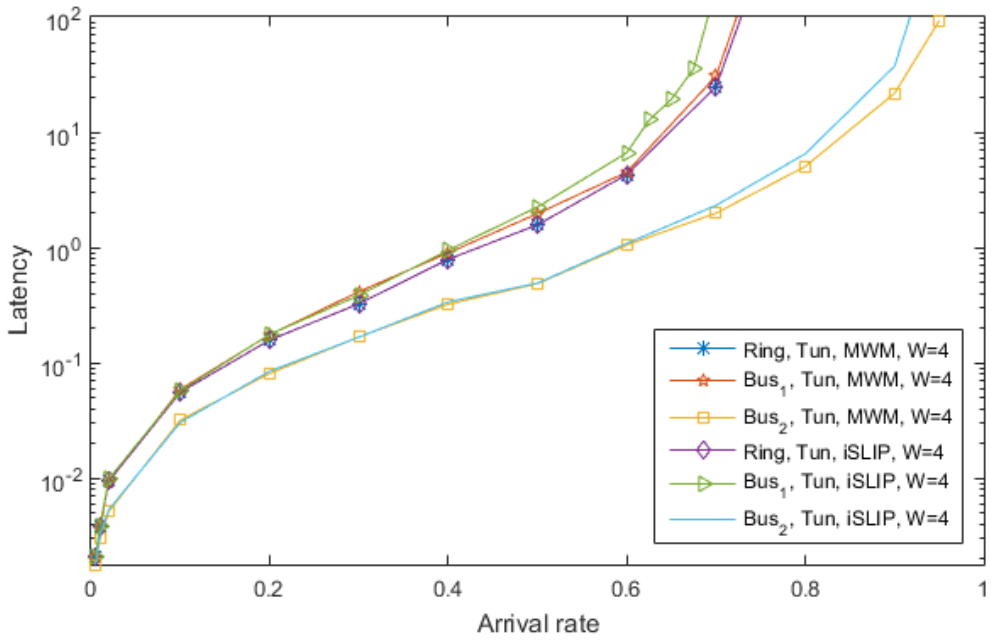

Figure 28: Tunable transmitters: latency vs. load for $W = 4$ and first step $iSLIP$ or MWM

5 Conclusions and Future Works

This thesis investigated the optimal scheduler for $ONoC$ and evaluated the scheduler performance through simulations for bus and ring topologies. The scheduling problem is characterized as bipartite graph matching problem. A two-step scheduling framework using MWM and $iSLIP$ in the first step is propsed and evaluated.

Results shown that the ring topology performance is better than the bus topology with one transmitter per node.

Solutions for further improving the performance includes using the unidirectional bus with two transceivers per port and enabling transmitter tunability especially when the wavelengths are few. These solutions however comes at higher cost and complexity.

When the number of wavelength equals to half of the nodes, tunable architectures performance are better than the fixed architectures. Advantages shown that increasing the number of wavelengths leads to decrease the latency while increases throughput.

The MWM scheduler performance is better than the $iSLIP$ scheduler performance for both fixed and tunable transmitter architectures and for all type of topologies. However the MWM scheduler is complex and its good performance and stability come at the expense of high computation complexity.

Overall $iSLIP$ shows limited performance degradation and is able to combine high performance with low computational complexity.

Based on this work further studies can be done regarding the evaluation of performance of any traffic patterns. Also alternative topologies supporting bidirectional transmissions can be considered and assessed. In addition to that other scheduling frameworks can be considered.

References

[1] Peter Kogge, Ed., "Exascale computing study: Technology challenges in achieving exascale systems," Darpa IPTO, Tech. Rep., Sep. 2008.

[2] L. Barroso and U. Holzle, The Datacenter as a Computer: An Introduction to the Design ofWarehouse-Scale Machines. Morgan & Claypool Publishers, 2009.

[3] A. Benner, "Cost-effective optics: Enabling the exascale roadmap," in 17th IEEE Symposium on High Performance Interconnects (HOTI), Aug. 2009, pp. 133–137.

[4] J. Rehr, F. Vila, J. Gardner, L. Svec, and M. Prange, "Scientific computing in the cloud," Computing in Science Engineering, vol. 12, no. 3, pp. 34–43, May-June 2010.

[5] W. Dally and B. Towles, Principles and Practices of Interconnection Networks. San Francisco, CA, USA: Morgan Kaufmann, 2003.

[6] H. J. Chao and B. Liu, High Performance Switches and Routers. Wiley- IEEE Press, 2007.

[7] N. Farrington, G. Porter, S. Radhakrishnan, H. H. Bazzaz, V. Subramanya, Y. Fainman, G. Papen, and A. Vahdat, "Helios: a hybrid electrical/optical switch architecture for modular data centers," SIGCOMM Comput. Commun. Rev., vol. 40, pp. 339–350, Aug. 2010.

[8] M. Petracca, B. Lee, K. Bergman, and L. Carloni, "PhotonicNoCs: System- Level Design Exploration," IEEE Micro, vol. 29, no. 4, pp. 74–85, Jul.-Aug. 2009.

[9] X. Chen, L.-S. Peh, G.-Y. Wei, Y.-K. Huang, and P. Prucnal, "Exploring the design space of power-aware opto-electronic networked systems," in International Symposium on High-Performance Computer Architecture (HPCA), Feb. 2005, pp. 120–131.

[10] D. A. B. Miller, 'Rationale and Challenges for Optical Interconnects to Electronic Chips', Proceedings of the IEEE, vol. 88, no. 6, pp. 728-749, 2000.

[11] I. P. Kaminow, "Optical integrated circuits: A personal perspective," Journal of Lightwave Technology, vol. 26, no. 9, pp. 994–1004, May 1 2008.

[12] A. F. Benner, M. Ignatowski, J. A. Kash, D. M. Kuchta, and M. B. Ritter, "Exploitation of optical interconnects in future server architectures," IBM Journal of Research and Development, vol. 49, no. 4/5, pp. 755–775, July 2005.

[13] R. S. Tucker, "The role of optics and electronics in high-capacity routers," Journal of Lightwave Technology, vol. 24, no. 12, pp. 4655 – 4673, Dec. 2006.

[14] A. Gnauck, G. Charlet, P. Tran, P.Winzer, C. Doerr, J. Centanni, E. Burrows, T. Kawanishi, T. Sakamoto, and K. Higuma, "25.6-Tb/s WDM Transmission of Polarization-Multiplexed RZ-DQPSK Signals," Journal of Lightwave Technology, vol. 26, no. 1, pp. 79–84, Jan. 1 2008.

[15] V. Soteriou and L.-S. Peh, "Exploring the design space of self-regulating power-aware on/off interconnection networks," IEEE Transactions on Parallel and Distributed Systems, vol. 18, no. 3, pp. 393–408, Mar. 2007.

[16] A. Bianco, D. Cuda, M. Garrich, R. Gaudino, G. Gavilanes, P. Giaccone, and F. Neri, "Optical interconnection networks based on microring resonators," in IEEE International Conference on Communications (ICC 2010), May 2010.

[17] P. Castoldi, P. G. Raponi, N. Andriolli, I. Cerutti, and O. Liboiron- Ladouceur, "Energy-efficient switching in optical interconnection networks," in 13th International Conference on Transparent Optical Networks (ICTON 2011), Stockholm, Sweden, June 26–30 2011, pp. 1–4.

[18] Mekkittikul, Adisak and McKeown, Nick, "A practical scheduling algorithm to achieve 100% throughput in input-queued switches",IEEE INFOCOM, vol. 2 pp. 792-799, 1998.

[19] N. McKeown, A. Mekkittikul, V. Anantharam, and J.Walrand, "Achieving 100% throughput in an input-queued switch," IEEE Transactions on Communications, vol. 47, no. 8, pp. 1260–1267, Aug. 1999.

[20] Tabatabaee, Vahid and Tassiulas, Leandros, "MNCM: A critical node matching approach to scheduling for input buffered switches with no speedup", IEEE/ACM Transactions on Networking, pages 294–304,2009,IEEE

[21] P. G. Raponi, N. Andriolli, I. Cerutti, and P. Castoldi, "Two-step scheduling framework for space-wavelength modular optical interconnection networks," IET Communications, vol. 4, no. 18, pp. 2155–2165, December 17 2010.

[22] N. Mckeown 'Scheduling algorithms for input-queued cell switches', PhD dissertation, UC Berkeley, May 1995.

[23] L. Tassiulas and A. Ephremides, "Stability properties of constrained queueing systems and scheduling policies for maximum throughput in multihop radio networks," IEEE Transactions on Automatic Control, vol. 37, no. 12, pp. 1936–1948, Dec. 1992.

[24] Develder, Chris and Cheyns, Jan and Van Breusegem, Erik and Baert, Elise and Colle, Didier and Pickavet, Mario and Demeester, Piet, "Architectures for optical packet and burst switches", ECOC, vol. 4, no.1, pp 100-103.

[25] Scicchitano, Alessandra and Bianco, Andrea and Giaccone, Paolo and Leonardi, Emilio and Schiattarella, Enrico "Distributed scheduling in input queued switches", IEEE International Conference on Communications pp 6330-6335, 2007.

[26] Bayati, Mohsen and Prabhakar, Balaji and Shah, Devavrat and Sharma, Mayank, "Iterative scheduling algorithms", IEEE INFOCOM, pp 445-453, 2007.

[27] Li, Xike and Elhanany, Itamar, "Stability of a frame-based oldest-cell-first maximal

weight matching algorithm", IEEE Transactions on Communications, vol. 56, no.1, pp 21-26. 2008.

[28] J. Dai and B. Prabhakar, "The throughput of data switches with and without speedup," in IEEE INFOCOM, vol. 2, Mar. 2000, pp. 556–564.

[29] D. Shah, P. Giaccone, and B. Prabhakar, "Efficient randomized algorithms for input-queued switch scheduling," IEEE Micro, vol. 22, no. 1, pp. 10–18, Jan/Feb 2002.

[30] P. Pintus, P. Contu, P. G. Raponi, I. Cerutti, and N. Andriolli, "Silicon-based all-optical multi microring network-on-chip," Opt. Lett., vol. 39, no. 4, pp. 797–800, Feb. 2014.

[31] F. Gambini, P. Pintus, S. Faralli, N. Andriolli, and I. Cerutti, "A photonic integrated network-on-chip with multi microrings," in OFC/NFOEC Tech. Dig., Mar. 2015.

[32] R. K. Ahuja, T. L. Magnanti, and J. B. Orlin, Network Flows: Theory, Algorithms, and Applications. Prentice Hall, 1993.

[33] H. Zmg, J. P. h e , and B. Mukherjee. ' AR eview of Routing and Wavelength Assign-niea Approaches for Wavelength-Routed Optical WDM Networks:' SPIWBallzer Optical Networks Magazine (ONM), vol. I, no. I, Jan. 2000.

Index