Costs and Benefits of Flexibility in Spatial Division Circuit Switched Networks-on-Chip

Ahsen Ejaz
Royal Institute of Technology, Sweden
ahsen@kth.se

Axel Jantsch
Royal Institute of Technology, Sweden
axel@kth.se

Abstract

Although most Network-on-Chip (NoC) designs are based on Packet Switching (PS), the importance of Circuit Switching (CS) should not be underestimated. Many MPSoC executing real-time applications require an underlying communication backbone that can relay messages from one node to another with guaranteed throughput. Compared to PS, CS can provide guaranteed throughput with lower area and power overheads. It is also highly suited for applications where nodes transfer long messages. Spatial Division Multiplexing (SDM) can allow more efficient use of available network resources by dividing them among multiple simultaneous transactions. The network developed by Vali [1] has three design variations based on the number of sub-channels, has a predictable connection setup time, and uses CS to provide guaranteed throughput once a connection is established. In this paper we use this network as a basis to study the effect of flexibility based on SDM, on the performance of CS networks. A network evaluation platform has been developed to configure and evaluate networks with a maximum of 8 sub-networks, with each sub-network comprising of 1, 2 or 4 sub-channels. We show that under uniform traffic pattern and requests of uniform random bandwidth (BW) requirement, a less flexible network outperforms a network with higher flexibility due to a phenomenon we call ‘stray requests’. We conclude this paper by showing that under high network traffic, performance of our flexible networks can be as much as 113% better than HAGAR [2] and Liu’s [3] network.

Categories and Subject Descriptors


1. INTRODUCTION

The advantages of a PS network over a CS network and vice versa depend on the application which uses them. PS NoCs are usually well suited for non-real-time systems where the network provides Best Effort Services (BES) [4]. Nodes in such systems need to transfer small infrequent bursts of data between each other [5]. In contrast, CS NoCs are more suited for real-time applications which require Guaranteed Services (GS) in terms of maximum latency and guaranteed throughput [4]. A CS NoC reserves an exclusive path between source and destination, at the start of communication, over which messages can propagate independent of traffic in the network and without need for transmission of any subsequent routing information or processing of individual message flits. As a result, CS NoCs have a lower latency for larger message sizes (lasting for 1 or more seconds) [5] e.g. in the case of a multimedia application relaying a stream of video or audio data at a fixed bitrate. In [6], Liu shows a detailed analysis that CS networks will outperform PS NoCs for packet size exceeding 30 flits/packet under low load and 60-70 flits/packet under high network load.

Although PS NoCs like Nostrum [7] and SuperGT NoC [8] provide GS, they require implementation of additional protocols which increases network complexity, area and power consumption. This limitation of PS NoC is overcome by hybrid networks which use PS to provide BES and CS to offer GS, e.g. hybrid NoC by Lusala and Legat [4] and Æthereal NoC [9]. The dual nature of these networks complicate the design of their Resource Network Interfaces (RNI) which assists communication between the resource and network router. To simplify both the network and RNI design, network presented in [3] is based on only CS. It provides GS but is not suitable for applications which require BES.

Moreover, contention control in PS networks is provided by using priorities associated with individual packets to drop, misroute, or buffer some contending packets while forwarding others. Such characteristics of a PS NoC make its use unappealing in an area and power constrained environment [6, 10]. On the other hand, flits travelling on a reserved path in a CS network do not face any contention leading to a simplified and energy efficient design.

In PS NoCs, where Time Division Multiplexing (TDM) is used to divide available BW among multiple simultaneous messages, SDM can be used in CS NoCs to divide wires between nodes (channels) into groups (sub-channels) and allocating them to concurrent data streams depending upon their BW requirements. This allows more efficient utilization of the network BW. Unlike TDM, where configuration of a switch has to be updated after every timeslot, for a SDM based CS network, it remains stable until the connection is terminated. This improves power consumption of the network.

In spite of the above mentioned advantages of CS networks over PS networks, their properties are not fully understood and need further exploration. This paper presents a novel CS NoC which is an improvement over previous designs and investigates the effect of flexibility in the SDM [5] based NoC. We hypothesize that network performance, in terms of its request acceptance rate and connection setup time, increases with increase in flexibility. The network employs an exhaustive parallel search algorithm [11] to reserve a path, from source to destination, in predictable time. We test the network using ‘retry until success’ and ‘no retry’ programming models as presented by Liu [3], but with some alteration.

We uncover a phenomenon called ‘stray requests’, which causes a less flexible network to outperform a network with higher flexibility when operated at the same clock frequency. Taking into account the maximum clock frequencies of the networks, we discover that a network with lower flexibility will always have higher performance.

Section II presents relevant research done by others and shows how our work fits in. In section III the programming models, networks and
the evaluation platform are presented. Section IV gives synthesis results of various SDM based CS NoC configurations. Performance results are presented in section V. Advantages of a more flexible CS NoC over HAGAR and Liu’s NoC are presented in section VI.

2. RELATED WORK
SDM in CS NoC is still an untapped research area where little work has been done. Leroy et al. [5] have developed an SDM based CS NoC which provides GS and uses misrouting to setup connections [12]. They implement a multistage interconnection network based on rearrangeable non-blocking Beneš switch which reduces the area of the switch compared to a full-crossbar switch, but increases the complexity of the switch controller. To provide maximum switching flexibility, internal switch re-routing is required which further complicates the switch controller. Our most flexible network uses a 16×20 full-crossbar switch and minimal path routing which simplifies the switch controller and eliminates the need for re-routing after a connection is established.

Wolkotte et al. [13] have implemented a CS NoC using Lane Division Multiplexing which is similar to SDM. They use a separate Best Effort (BE) network to transmit control information used to configure routers and setup connections. A similar SDM based CS NoC with PS designed by Lusala and Legat [14] also uses a BE setup packet, transmitted over PS sub-network, to reserve a path over the CS network. In our opinion, using two networks, one to setup the other is redundant. In our network connection setup probes propagate over free channels of the CS network to configure routers. This utilizes network resources more efficiently and when combined with parallel search probing [11], it makes connection setup time predictable under the no retry programming model.

In [2], authors present a non-SDM based NoC which provides GS using BES PS network to propagate the connection request fits to a centralized node called NoCManager. It uses an innovative technique called HAGAR to reserve a path from source to destination. Use of BES to setup a GS connection makes its setup time unpredictable and renders network not scalable [2, 3].

A non-SDM based network designed by Liu [3] is quite similar to the network we have used. Other than SDM, the main difference is that it uses priorities and preemption to overcome contention while in our network a request contending for an output channel, reserved by another prior request which is also in setup phase, always gets rejected. Using preemption complicates Liu’s NoC and requires use of dual clock to improve performance. Our network using a simpler design shows higher performance, as is presented in section VI.

None of the above cited publications provide a comparison of performance of networks with different levels of flexibility. In this paper our contribution is an experimental comparison of CS networks with different levels of SDM based flexibility and investigation of the underlying causes of performance deterioration in a network of higher flexibility. We use a SDM based CS network which not only offers guaranteed throughput but also has a predictable connection setup time making it suitable for use with real-time applications.

3. EXPERIMENTAL PLATFORM

3.1. Programming Models
The network is simulated and results are compiled under two different programming models.

In the retry until success model, once a Processing Element (PE), which can be an IP core connected to the network via RNI, forwards a connection setup request to its RNI, it will receive a pending response from the RNI until the request is successfully accepted by the network. When connection is established, RNI transmits an ACK signal to the PE after which it commences data transmission. However, if the RNI receives a rejection response from the network due to unavailable network resources, then the request will be retried after a pseudo-random time interval. This scheme helps avoid request livelocks and simplifies router design compared to Liu’s design which uses priorities and preemption [3].

Under the no retry programming model [3], if the RNI receives a request rejection response from the network, it is transmitted to the PE and that request is considered to have failed and not retried by the RNI. The PE, if it so desires, can forward the same request to the RNI again. If successful, a connection is setup in exactly $4 \times (D + 1) + 5$ cycles (D is the source to destination hop count, each node requires 3 cycles to forward a request and 1 cycle to propagate the ACK signal upstream, 1 cycle is spent in the input queue, 3 cycles by RNI-transmitter at source and 1 cycle by RNI-receiver at destination), otherwise it will be rejected in shorter time. Contrary to retry until success model, this programming model is more suitable for safety critical applications where the requirement of predictability is more emphasized than guaranteed throughput.

3.2. Network
Each router, of the mesh network, has 5 ports, 4 to connect to its neighbors and 1 to connect to a resource. Network has three design variations based on SDM i.e. single, dual and quad channel with 1, 2 and 4 (input and output) sub-channels per port respectively, as shown in Figure 1. For a multichannel router (Figure 1b, 1c), an incoming sub-request can be allocated to any idle output sub-channel, of equivalent BW, in the required direction. This saves link BW and increases path diversity. This flexibility, however, comes at a cost of increased router complexity and higher switch area.

![Figure 1: SDM based network routers (a) single channel, (b) dual channel and (c) quad channel](image1)

Each sub-channel has at least 32 data lines, 2 Valid-Request (VR) control lines and 2 ACK response lines. Data lines propagate connection requests in the setup phase, data during the data phase and are idle otherwise. VR control lines propagate control signals, encoding the type of word on the data lines. ACK response lines carry response (idle, pending, connection accepted, rejected) in the direction opposite to data lines, from the destination to the source.

A simplified single channel router is shown in Figure 2. When a connection request appears at the input of Input Channel Controller (ICC), it sends a Pending response upstream and forwards the
request to the Router Controller (RC). RC rejects duplicate request, translates the destination of a requests into its direction of output, performs arbitration and allocates idle output sub-channels to incoming requests. If non-duplicate requests are competing for the same output channel then we use static priority such that the priority of resource -> west -> east -> south -> north. Only winning requests will propagate in both X and Y direction while others are rejected. RC sets control registers in the Switch Controller (SC) which configures the switch and connects a particular input to an output.

Crossbar switches are implemented using multiplexers instead of Beneš switches which requires more area [1]. By taking into account the fact that an incoming request from a particular direction will not be routed back in the same direction, we optimize 5x5, 10x10 and 20x20 switches into 4x5, 8x10 and 16x20 crossbars switches for single, dual and quad channels routers, respectively. Due to greater level multiplexing, higher order switches have higher area and timing requirements.

The network uses parallel search probing, which is an exhaustive algorithm to find a minimal path from source to destination in a predictable time. For a source to destination hop count of n, if a free path exists, a connection will be setup in 4n clock cycles, otherwise it will be rejected in less time. Devisagamani et al [11] have presented this concept in its entirety. It is briefly explained below.

The data transmission has two phases, a connection setup phase and a data phase. As shown in Figure 3, when Node (0, 0) wants to setup a connection with node (2, 1), it transmits a connection request in all directions leading to the destination through a minimal path. A connection request consists of 14 bits for each of the source and destination addresses, 2 bits to distinguish between sub-requests from the same source node and 2 bits to represent the number of sub-channels being used. Each node requires 3 clock cycles to forward a request to the next node if free channels are available, otherwise a rejection signal is propagated upstream.

![Figure 3: Connection setup in 2x3 network using exhaustive parallel search algorithm](Image)

Both nodes (0, 1) and (1, 0) upon receiving connection requests forward them to the next node along the minimal path. Node (1, 1) will simultaneously receive the requests from north and west. If duplicate requests are received, as in this case, a Duplicate Checker at each node will give priority to the request from horizontal sub-channels (east and west) over a request from vertical sub-channels (north and south). Consequently, node (1, 1) will accept incoming request from (0, 1) forwarding it to node (2, 1) while rejecting the request from (1, 0). A similar procedure is followed at node (2, 1).

Once a request has reached the destination, a request accepted signal is transmitted from (2, 1) to (0, 0) which travels at 1 hop per cycle through nodes (1, 1) and (0, 1). After receiving the connection accepted signal, the source node can initiate data transmission. On its completion, a connection termination request is transmitted by the source node which propagates at 1 hop per cycle and frees reserved network resources on its way to the destination.

### 3.3. Resource Network Interface (RNI)

The RNI connects to the resource input and output sub-channels of the routing node on the network side and to the IP core on resource side as shown in Figure 4. It has two main parts, a transmitter and a receiver. The transmitter divides an incoming request into sub-requests based on its BW requirement and the available BW of network sub-channels and then transmits the sub-requests on individual sub-channels. A transmitter forwards a request to the interconnect after 3 clock cycles and then it waits for a response. An ACK response is directly conveyed to the requesting IP core.

The behavior on a request rejection response depends on the used programming model. A partially accepted request (when some sub-requests are rejected while other are being processed or accepted) is considered rejected and is retried or canceled depending on the programming model. The non-rejected parts of the request are prematurely terminated. If possible, on retry, the transmitter avoids the sub-channels over which a rejection was received. After data transmission is complete the transmitter tears down the connection.

![Figure 4: Interconnection of various system modules](Image)

When a connection request appears on the input of an RNI-receiver, it first verifies if it is received in its entirety using sub-channels occupied count transmitted with each sub-request. Since all sub-requests propagate at the same rate through the network, they will reach the destination simultaneously. A request that partially reaches a destination is rejected by the receiver because it will not fulfill the requested BW requirement. If two or more complete requests simultaneously reach the receiver then the request with the bigger BW×Source-Destination-Distance product is accepted while others are rejected. If this product is the same then the request with highest source ID (x-coordinate + row size × y-coordinate) is accepted. The receiver sends an accepted or rejected response upstream after 1 clock cycle. After accepting a request, a receiver waits to receive a request ID from the source node. This informs the receiver of the order in which incoming sub-requests are located over the sub-channels and is used to correctly assemble incoming data. It is then forwarded to the receiving core (slave).

### 3.4. Traffic Generation and Measurement

The evaluation platform is implemented in an open loop configuration, as shown in Figure 4, where the behavior of request generation unit is not affected by the performance of the network.

A random request generator implemented in SystemVerilog is used to generate connection requests for the network under evaluation. Each request has a lifetime (Poisson distribution), required BW, destination address (based on traffic pattern), a request ID and is separated from consecutive requests by a random time period (Poisson distribution). All generated requests enter into an infinite queue which isolates the request generator from the RNI.

We simulate networks with 4 different levels of flexibility as shown in Figure 5. Networks are encoded in NxCxB format, such that N is the number of sub-networks in the network, C is the number of sub-channels per sub-network and B is the number of bits per sub-channel e.g. 4x1x32 has four single channel 32 bit sub-networks.

In terms of flexibility these network configurations are ranked as 1x1x128, 4x1x32, 2x2x32 and 1x4x32 with 1x1x128 network being the least flexible. Even though 4x1x32 and 2x2x32 networks have 4 and 2 sub-networks respectively, they still rank below 1x4x32 network because in multi-network networks request in one sub-network cannot use free sub-channels of the other sub-network.
Synopsys Design Compiler using TSMC 90 nm library optimized node transmits data of a measurable request [3].

Route rate is the fraction of traffic acceptance time over which a network to the successful establishment of a connection. Getting accepted by the RNI-transmitter for transmission over the connection. Request setup time is the time interval from a request at the generator to successful establishment of its required connection requests, indicate that 4x1x32 network outperforms others in terms of average accepted IR, request grant time and request setup time, as shown in Figure 7. After 4x1x32, other networks can be ranked in order as 1x1x128, 2x2x32 and 1x4x32. The observed results invalidate our hypothesis. Increased flexibility in SDM based CS network does not guarantee higher performance.

5. Experimental Results

Each network and its RNI was simulated at the network’s respective clock frequency as is shown in Table 1. All network nodes generate uniform random traffic with average connection lifetime of 200 cycles. The request BW has a uniform random distribution over 32, 64, 96, 128 bits/cycle, unless mentioned otherwise. Evaluation results obtained using Cadence NCSim simulator are as follows.

5.1. Retry Until Success Programming Model

The observed results for a 4x4 network, generating 48000 connection requests, indicate that 4x1x32 network outperforms others in terms of average accepted IR, request grant time and request setup time, as shown in Figure 7. After 4x1x32, other networks can be ranked in order as 1x1x128, 2x2x32 and 1x4x32. The observed results invalidate our hypothesis. Increased flexibility in SDM based CS network does not guarantee higher performance.

Traffic generation (acceptance) time is defined as the time interval from the generation of the first measurable request at any source node of the network to the generation (acceptance) of the last measurable request plus its lifetime, as shown in Figure 6.

Figure 6: Traffic generation time and traffic acceptance time

Offered Injection Rate (IR) per master node is defined as the ratio of total measurable requests generated per master node to traffic generation time. Similarly, accepted IR per node is defined using traffic acceptance time in place of generation time. Offered (accepted) data rate per node is ratio of total data transmitted by measurable request/node to traffic generation (acceptance) time.

Request grant time is the time interval from the generation of a request at the generator to successful establishment of its required connection. Request setup time is the time interval from a request getting accepted by the RNI-transmitter for transmission over the network to the successful establishment of a connection.

Route rate is the fraction of traffic acceptance time over which a node transmits data of a measurable request [3].

\[
\text{Route rate} = \frac{\text{Requests per node} \times \text{Avg. lifetime}}{\text{Traffic acceptance time}}
\]

4. SYNTHESIS RESULTS

All three network variations (single, dual and quad channel networks) and configurations (1x1x128, 4x1x32, 2x2x32 and 1x4x32) with channel BW of 128 bits/cycle were synthesized with Synopsys Design Compiler using TSMC 90 nm library optimized for maximum clock frequency. Results are shown in Table 1.

Table 1: Synthesis Results For A Switch Of Different Networks With 128 Bits/Cycle Total Channel Bandwidth

<table>
<thead>
<tr>
<th>Network Type</th>
<th>Single Channel (1x1x128)</th>
<th>Dual Channel (1x2x64)</th>
<th>4x1x32</th>
<th>2x2x32</th>
<th>Quad Channel (1x4x32)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Area (um2)</td>
<td>35086</td>
<td>58444</td>
<td>71024</td>
<td>87320</td>
<td>132768</td>
</tr>
<tr>
<td>NAND gates/bit</td>
<td>122.80</td>
<td>204.55</td>
<td>248.58</td>
<td>305.62</td>
<td>464.69</td>
</tr>
<tr>
<td>Power (mW)</td>
<td>30.1</td>
<td>32.6</td>
<td>62.2</td>
<td>44.42</td>
<td>36.12</td>
</tr>
<tr>
<td>Clock (GHZ)</td>
<td>2.1</td>
<td>1.52</td>
<td>2.1</td>
<td>1.52</td>
<td>1</td>
</tr>
</tbody>
</table>

Figure 7: Evaluation results for network at maximum clock frequency shown in Table 1

For all networks, average request grant time increases exponentially until the saturation request IR and then it becomes unbounded because of increasing number of requests in the infinite source queue. Average request setup time stabilizes after reaching a maximum value at saturation IR. Most flexible 1x4x32 network has highest request grant and setup times due to its slow clock.

The main contributing factor behind the invalidation of our hypothesis is the drastic difference in maximum clock frequencies of evaluated network configurations. Even for a simple case, where all the requests have 32 bits/cycle bandwidth requirement (1 network sub-channel), performance-wise networks are still ranked in the same order as for uniform random bandwidth requirement.

When these networks are evaluated at the same clock frequencies, we still observe that higher SDM based flexibility does not guarantee higher performance in terms of accepted request IR. Figure 8a shows that under uniform random request BW and clock frequency of 1 GHz, all networks outperform 1x1x128 network and the 1x4x32 network has highest accepted request IR. The 2x2x32 network however, lags behind a less flexible 4x1x32 network.

If all requests have a BW of 32 or 64 bits/cycle, we observe that a network with higher SDM based flexibility has higher performance as shown in Figure 8b and c respectively. If request BW is fixed to 96 bits/cycle for all requests, then the least flexible network, 1x1x128, has highest accepted request IR as shown in Figure 8d.

This is due to a phenomenon explained in Figure 9. Consider a portion of 4x1x32 network serving a 96 bits/cycle BW request, between nodes 4 and 2 via nodes 1 and 5. Now, node 3 wants to establish a 96 bits/cycle connection with node 5 and forwards a request on 3 sub-channels. Upon reaching node 4, a rejection response will be sent upstream on one of the three sub-channels, while other two will propagate forward. These two sub-requests, even if successful, will not fulfill the BW requirement of node 3. We call these sub-requests stray requests. When the rejection response of the rejected sub-request reaches the RNI of node 3, it will send a premature connection request termination signal (travelling at 1 hop/cycle) to remove remaining two sub-request
from the network. By the time it is propagated, stray requests would have reached the receiver (slave) of node 5. These sub-channels will remain reserved until their sub-requests are either rejected due to contention at an intermediate node or by the receiver of the destination due to an incomplete request arrival or prematurely terminated by the termination signal sent from the source node.

The advantage of using one network over another is highly dependent on the traffic pattern and request BW distribution. Under uniform random traffic a quad channel network will have best performance except when average request BW to channel BW ratio is between 0.75 and 1, in which case non-SDM based network will show higher performance (Figure 8d). Based on other experiments (not presented here) increased flexibility will result in higher performance under neighbor and tornado traffic patterns [15]. But even if the performance of a higher flexibility SDM based network is greater than others, it should only be implemented if benefits of its higher performance outweigh its higher area and power costs.

5.2. No Retry Programming Model

Obtained results for a 6x6 network generating 108000 connection requests, shown in Figure 10, indicate that a 4x1x32 network outperforms other networks for lower route rate (below 0.4), while for higher route rates 1x4x32 network has higher connection success rate. Higher performance of 4x1x32 network was expected (as it is for lower route rate) since quad and dual channel networks can propagate stray requests deeper into the network, which in turn leads to more connection rejections, stray request generation and performance degradation in those networks. Intuitively, this trend (higher performance of less flexible networks) should have continued for the whole range of route rates but this is not the case.

Stray requests waste network BW by unnecessarily holding sub-channels. They can exist in all multichannel network configurations depending on the BW of connection requests. However, no stray requests are generated in non-SDM based networks making its performance independent of the request BW. For SDM based networks, if all requests are of either 32, 64 or 128 bits/cycle bandwidth, no stray requests are generated (due to unavailability of sub-requests in case of 32 bits/cycle, both sub-requests get rejected in case of 64 bits/cycle, and unavailability of free sub-channels for 128 bits/cycle requests). Stray requests can be avoided if each network node only forwards a request when all its sub-requests can also be forwarded in a particular direction. But this will not fully exploit path diversity and SDM. It will be counterproductive.

If it is intended to use a network at its maximum operating frequency, then the simplicity of a network based on combination of multiple single channel networks (e.g. 4x1x32) will enable it to operate at the highest frequency (compared to dual and quad channel networks) and always outperform other SDM based network configurations. Under uniform random traffic pattern and a distribution of request BW requirement such that average request BW to channel BW ratio is between 0.75 and 1, a simpler single channel network (1x1x128) can marginally outperform a network with multiple single channel sub-networks. This is because they both can have the same clock frequency and the performance of SDM based networks will be hindered by stray requests (Figure 8d).

If the networks are operated at same clock frequency, then the advantage of using one network over another is highly dependent on the traffic pattern and request BW distribution. Under uniform random traffic a quad channel network will have best performance except when average request BW to channel BW ratio is between 0.75 and 1, in which case non-SDM based network will show higher performance (Figure 8d). Based on other experiments (not presented here) increased flexibility will result in higher performance under neighbor and tornado traffic patterns [15]. But even if the performance of a higher flexibility SDM based network is greater than others, it should only be implemented if benefits of its higher performance outweigh its higher area and power costs.
If we take maximum operating frequencies of networks in to account then the least flexible SDM based network (e.g. 4x1x32) will always perform better despite of its lower connection success rate, because firstly requests will be served in shorter time and secondly, the resource can retry a rejected request at a higher speed, effectively increasing connection success rate.

No retry programming model exploits predictable response time of the network leading to a constant maximum connection setup time of 49 cycles, independent of network traffic. This improves isolation between computation and communication resource, enhances compositionality and makes the system more predictable.

If we use a receiver (slave) that is capable of handling multiple simultaneously incoming requests, then whereas the saturation IR under retry until success model and connection success rate under no retry model increases for multichannel networks, their relative rank, in terms of performance, remains the same [15].

6. Comparison with Other Networks

Liu’s NoC [3] uses a dual clock scheme with control and data path frequencies of 570 MHz and 1.8 GHz respectively and requires 292 NAND gates/bit, when synthesized with SMIC 90 nm library [3]. On the other hand, 8x8 and 16x16 HAGAR networks have clock frequencies of 200 and 50 MHz respectively, when synthesized using FARADAY’s 130nm UMC Library [2] (288 and 72 MHz respectively, when scaled to 90nm technology using general scaling [16]). The sequential probe switch of HAGAR network requires 778 NAND gates [3]. Our network node has higher clock frequencies which are same for both control and data path and independent of network size and has lower area requirements for single and dual channel network variations as shown in Table 1.

Performance comparison of Liu’s NoC with our multichannel 16x16 network with 50% master ratio (uniform random master node distribution), with average connection lifetime of 200 cycles and no retry programming model, is shown in Figure 12. Multichannel networks perform better than Liu’s and HAGAR NoC when connection success rate is less than 60% and 80% respectively, because lower clock frequency of Liu’s network, particularly in the setup phase, degrades network performance under high network traffic. HAGAR network suffers because it uses best effort flits to forward connection request packets to a centralized NoCManager to reserve a path. The PS sub-network also does not efficiently exploit path diversity as our network does using parallel search probe. For a route rate greater than 0.5, a quad channel network outperforms the HAGAR NoC by 80% to 113% in terms of connection success rate. Similar results were obtained for master ratio of 20%. The effective connection success rate is much higher if we also take higher clock frequencies of single and dual channel based SDM networks (e.g., 4x1x32, 2x2x32).

We discover that increased SDM based flexibility comes at the cost of increased network area, complexity and lower clock frequency, while performance gains are highly dependent on network traffic. When operated at maximum frequency, subject to network traffic, either a non-SDM based network or the least flexible SDM based NoC will outperform others due to their higher clock frequencies.

At the same operating frequency, a less flexible network can outperform a more flexible network depending on network traffic. This is because various network configurations react to different network traffic differently and a diverse amount of generated stray requests can cause a variation in performance ranking of networks.

Under the no retry programming model, with same operating frequency, uniform random traffic pattern and request BW distribution, a more flexible NoC performs better than a less flexible NoC under high route rate at the cost of lower average source to destination distance and average BW of accepted requests. Under high traffic, the networks also have a higher connection success rate and are in general an improvement over Liu’s and HAGAR NoC. When operated at maximum frequency, depending on network traffic, either a single channel network or the least complex (flexible) SDM based network will always outperform others. A major advantage is predictable response and connection setup time.

8. References