



Deposited via The University of Leeds.

White Rose Research Online URL for this paper:

<https://eprints.whiterose.ac.uk/id/eprint/99906/>

Version: Accepted Version

Proceedings Paper:

Hassen, F and Mhamdi, L (2015) A Multi-Stage Packet-Switch Based on NoC Fabrics for Data Center Networks. In: 2015 IEEE Globecom Workshops (GC Wkshps). 2015 IEEE Globecom Workshops (GC Wkshps), 06-10 Dec 2015, San Diego, California, USA. IEEE, pp. 1-6. ISBN: 978-1-4673-9526-7.

<https://doi.org/10.1109/GLOCOMW.2015.7414039>

Reuse

Items deposited in White Rose Research Online are protected by copyright, with all rights reserved unless indicated otherwise. They may be downloaded and/or printed for private study, or other acts as permitted by national copyright laws. The publisher or other rights holders may allow further reproduction and re-use of the full text version. This is indicated by the licence information on the White Rose Research Online record for the item.

Takedown

If you consider content in White Rose Research Online to be in breach of UK law, please notify us by emailing eprints@whiterose.ac.uk including the URL of the record and the reason for the withdrawal request.

A Multi-Stage Packet-Switch Based on NoC Fabrics for Data Center Networks

Fadoua Hassen Lotfi Mhamdi
School of Electronic and Electrical Engineering
University of Leeds, UK
Email: {elfha, L.Mhamdi}@leeds.ac.uk

Abstract—Bandwidth-hungry applications such as Cloud computing, video sharing and social networking drive the creation of more powerful Data Centers (DCs) to manage the large amount of packetized traffic. Data center network (DCN) topologies rely on thousands of servers that exchange data via the switching backbone. Cluster switches and routers are employed to provide interconnectivity between elements of the same DC and inter DCs and must be able to handle the continuously variable loads. Hence, robust and scalable switching modules are needed. Conventional DCN switches adopt crossbars or/and blocks of memories in multistage interconnection architectures (commonly 2-Tiers or 3-Tiers). However, current multistage packet switch architectures, with their space-memory variants, are either too complex to implement, have poor performance, or not cost effective. In this paper, we propose a novel and highly scalable multistage packet-switch design based on Networks-on-Chip (NoC) fabrics for DCNs. In particular, we describe a novel three-stage packet-switch fabric with a Round-Robin packets dispatching scheme where each central stage module is based on a Unidirectional NoC (UDN), instead of a conventional single-hop crossbar fabric. The proposed design, referred to as Clos-UDN, overcomes all the shortcomings of conventional multistage architectures. In particular, as we shall demonstrate, the proposed Clos-UDN architecture: (i) Obviates the need for a complex and costly input modules, by means of few, yet simple, input FIFO queues. (ii) Avoids the need for a complex and synchronized scheduling process over a high number of input-output modules and/or port pairs. (iii) Provides speedup, load balancing and path-diversity thanks to a dynamic dispatching scheme as well as the NoC based fabric nature. Extensive simulation studies are conducted to compare the proposed Clos-UDN switch to conventional multistage switches. Simulation results show that the Clos-UDN outperforms conventional design under a wide range of input traffic scenarios, making it highly appealing for ultra-high capacity DC networks.

Keywords—Next-Generation Networking, DCN, Clos-network switch, NoC, packet dispatching, packet scheduling

I. INTRODUCTION

Large-scale workloads like the ones seen in Cloud computing, social networking and search impose stringent requirements on the data center switching fabric which needs support high capacity applications while being of low complexity, low latency and low power consumption. The network topology of a DCN describes how the switching modules and servers are weaved together. Fat-tree and folded Clos topology are commonly used to connect hosts to a set of aggregation switches and Top-of-Rack (ToR) switches. The ToR switches design typically requires high port densities to facilitate fast and modularized application deployments in DCNs. As scalability and reliability are inseparable, the switch design must

guarantee other features like high network throughput, low packets delay, path-diversity, load-balancing etc.

The traditional design approach consists of building hierarchical switching fabrics using switches and routers. Single-stage crossbar switch fabrics do not meet the growing networking requirements. While they can be implemented for small-sized switches, they become quite complex to implement and unscalable for growing port counts (beyond 64 ports) [1] [2]. Multistage switches design where many smaller crossbar fabrics are arranged in cascade have been typical commercial solutions for high-speed routers [3] mainly because they can be incrementally expanded by adding more modules to the existing design. Besides, they have numerous benefits such as being partially or completely non-blocking, providing good broadcast and multicast features and build in reliability with no or minimum failure in the system.

Three-stage Clos network that is frequently used for telecommunications and networking systems [4] [5]. As in single-stage, buffer placement defines the type of the multistage Clos switch. According to the stage where buffers are inserted, a Clos network could be a Space-Space-Space (S^3) [6] network without buffers or Memory-Memory-Memory (MMM) [7] with buffered switching units in all stages. Other combinations have also been studied [8] [9]. Despite their scalability potential, almost all existing Clos network based proposals are either too complex to be implemented, exhibit prohibitively high cost or have low performance. The input queuing structure at the input modules (IMs) is generally complex, requiring excessive number of queues to avoid the Head-of-Line (HoL) blocking [8] [7]. In addition to their impact on the scheduling complexity, these queues are generally required to be of output queued type and run much faster than the external input line rate. The scheduling (or dispatching) process in current multistage Clos networks, especially the MSM type, is very complex and expensive, yet has poor performance under non-uniform traffic scenarios. A typical example of this is the Concurrent Round Robin Dispatching (CRRD) for MSM and its enhanced versions [10] [11].

Irrespective of whether the switch is single- or multistage, it is constructed of either one or multiple single-hop crossbar fabrics as its elementary building block. In this context, the network-on-chip (NoC) paradigm has recently been gaining interest in modern high-performance single-stage switching fabrics design as it addresses a number of limitations of conventional single-hop crossbars, including scalability, port speed and path diversity [12]. A number of recent designs have used the NoC concept in high-performance switching.

A design for Ethernet switches has been described in [13] [14]. A Unidirectional NoC crossbar fabric based packet switch (UDN) design has been described in [12] [15] along with appropriate NoC routing algorithms. An extension of this design, termed Multidirectional NoC (MDN) packet switch has been also proposed in [16]. More recent results [17] proposed an implementation of a single-stage crossbar fabric using NoC-enhanced FPGA and different routing algorithms. Despite the high potential of NoC based crossbar fabrics, their application has been restricted to single-stage crossbar packet switches.

Motivated by the shortcomings of the previous switching architectures, we propose the first design of a multistage packet-switches based on NoC fabrics for DCNs. In particular, the proposed switching fabric is a combination of a Clos macro-design, that reports to the whole fabric architecture, and a UDN micro-design for the central switching modules of the packet-switch. We describe a three-stage Clos network with FIFO input queues and a dynamic dispatching of packets to the central modules. In the remainder of this paper we refer to the proposed multistage Clos-network switch fabric as the Clos-UDN. The proposed switching architecture has several advantages over earlier multistage packet-switches. In particular:

- The Clos-UDN obviates the need for a complex and costly input queueing structure. Unlike conventional multistage design where a high number of fast input queues (VOQs) is required, the Clos-UDN uses a small number of input FIFO queues which need not to run faster than the external line rate.
- The proposed Clos-UDN avoids the need for complex, costly and slow centralised scheduling process. Conventional multistage Clos networks require complex scheduling process with global synchronisation between inputs and outputs. Our proposal relies on the UDN stages to route input packets to their outgoing interfaces by means of fully distributed, parallel and independent NoC routers' decisions.
- The Clos-UDN inherits all the advantages of the UDN design in terms of scalability, speedup and path diversity [12]. These properties result in high performance in terms of low latency, high-throughput and efficient hardware design.

The remainder of the paper is structured as follows. Section II discusses relevant existing multistage Clos packet-switch architectures and their performance. In Section III, we describe the three-stage Clos-UDN packet-switch architecture, along with its NoC based central modules and its dispatching process. Section IV presents the performance study of the Clos-UDN switch and compares it to relevant existing Clos architectures. Finally, Section V concludes the paper.

II. RELATED WORK

Multistage network switches are more scalable than single stage crossbars. They are used in large-scale networks like DCNs for their scalability and reliability. They provide multiple routes between inputs and outputs, allowing the traffic to be balanced across alternative paths. Non-blocking Clos-network is a very popular design [4]. A three-stage Clos is generally

quoted as $\zeta(m, n, k)$ where m , n , and k are the parameters that completely define the structure of the network. The size of this Clos-network is N , where $N = n \times k$. The first stage is made of k input modules each of size $n \times m$. The middle stage has m switches each has k inputs and k outputs. Last, there are k output modules at the third stage, each of size $m \times n$. Extensive work has been done on Clos-network switches in all their variants such as S^3 [6], MSM [8] [11], SMM [9] and MMM [7] [2]. Unfortunately, none of the existing Clos-switching architectures has been shown to provide scalability in terms of cost, performance and hardware complexity. The MSM architecture requires expensive and complex input modules. Each of these input modules is required to cater for a high number of separate FIFO queues ($n.k$) in order to avoid the HoL blocking [7] [18]. Additionally, each of these queues is required to run $(n+1)$ times the line rate [10]. On the scheduling/dispatching front, the cost and practicality is a major issue. Two scheduling phases are required to resolve input-output ports contention. In addition to its high cost and long scheduling delays, no scheduling algorithm for this architecture has been shown to exhibit satisfactory performance [8] [11]. MMM [7] [2] mandate large and expensive internal memories to relax the scheduling process. Fully-buffered Clos architectures have good throughput performance since all contentions are absorbed by means of internal buffers. Although the scheduling process is better than that of MSM, it is still complex. Our work differs from all previously proposed architectures. We take a radically different approach at the heart of Clos-switch design by adopting NoC based fabrics as internal stages of the Clos-network. Designing each Central Module (CM) as a NoC brings a number of advantages that overcome the limitations of previous proposals. First, the input modules are less complex and cheaper compared to previous architectures. Each input module of the Clos-UDN switch requires only m input FIFO queues, each of which runs *twice* the line rate. This is to be compared to the MSM, for example, where each input module requires $(n.k)$ input FIFO queues each of which runs $(n+1)$ times the line rate. Contrary to the complex, costly and under-performing proposed schedulers in traditional Clos architectures, the Clos-UDN uses fully distributed and parallel scheduling at the NoC routers level, making it simple, fast and efficient as we shall describe next.

III. CLOS-UDN SWITCH ARCHITECTURE

This section describes the three-stage Clos-UDN switch architecture with NoC-based central modules. We describe the switch model with an emphasis on the NoC based central modules. We then introduce the dispatching process considered to transfer packets to the middle stage.

A. The switch model

The reference design of a Clos-UDN switch of size $N \times N$ is depicted in Fig. 1. The key notations used in this paper are listed as follows:

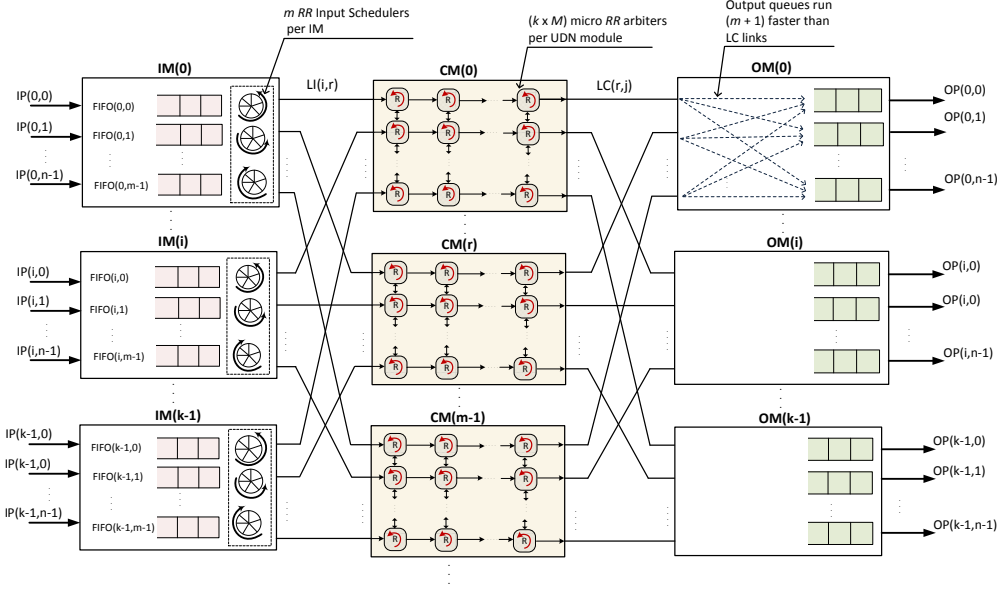


Fig. 1: $N \times N$ three-stage Clos-UDN packet-switch architecture with dynamic dispatching scheme

$IM(i)$	$(i + 1)$ th IM at the first stage
$CM(r)$	$(r + 1)$ th CM at the second stage
$OM(j)$	$(j + 1)$ th OM at the third stage
i	IM number, where $0 \leq i \leq k - 1$
r	CM number, where $0 \leq r \leq m - 1$
j	OM number, where $0 \leq j \leq k - 1$
h	IP/OP number in each IM/OM, respectively, where $0 \leq h \leq n - 1$
$IP(i, h)$	$(h + 1)$ th IP at $IM(i)$
$OP(j, h)$	$(h + 1)$ th OP at $OM(j)$
$FIFO(i, r)$	First-In-First-Out queue that stores packets going through CM module, r .
$LI(i, r)$	Output link at $IM(i)$ that is connected to $CM(r)$
$LC(r, j)$	Output link at $CM(r)$ that is connected to $OM(j)$

The third stage consists of k OMs, each of which has $m \times n$ dimension. Although it can be general¹, the proposed Clos-UDN architecture has an expansion factor $\frac{m}{n} = 1$, making it a *Benes* lowest-cost practical non-blocking fabric. An $IM(i)$ has m FIFOs each of which is associated to one of the m output links denoted as $LI(i, r)$. An $LI(i, r)$ is related to an $CM(r)$. Because $m = n$, each $FIFO(i, r)$ of an input module, $IM(i)$, is associated to one input port, $IP(i, h)$, and can receive at most one packet and send at most one packet to one central module at every time slot. A $CM(r)$ has k output links, each of which is denoted as $LC(r, j)$ and is connected to $OM(j)$. An $OM(j)$ has n OPs, each of which is $OP(j, h)$ and has an output buffer. An output buffer can receive at most m packets and forward one packet to the output line at every time slot. Packets destined to different output ports are accepted to the NoC fabric even when some outputs are busy with other packets.

¹The Clos-UDN can of course be of any size, where $m \geq n$. This would simply require packets insertion policy in the FIFOs should we need to maintain low-bandwidth FIFOs. We consider this to be out of the scope of the current work.

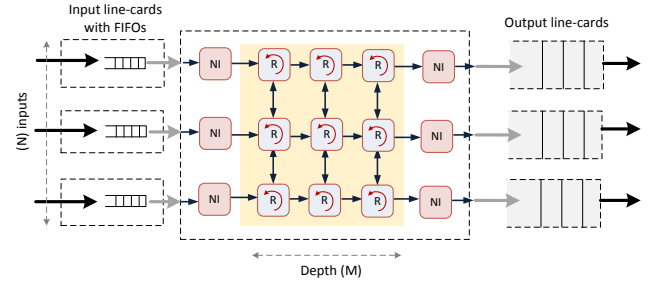


Fig. 2: The UDN crossbar switch

B. NoC based Central Modules

Our reference design is based on the UDN [12] fabric (Fig. 2) that we plug into the Clos central stages. In the Clos-UDN, every central unit is a two-dimensional mesh ($k \times k$) of small on-chip packet switched input-queued routers that transport packets across the NoC in a multi-hop fashion. All on-chip routers have small input FIFO queues of variable size (referred to as Buffer Depth- BD) to store packets on their journey to their outputs. To avoid elastic buffers, credit-based flow control is used and packets are only sent when buffer space is available [19]. A packet is of fixed-size with relative routing information stored at its header. Packets are fully received and stored in one of the router's buffers before going to the next hop. Using a deadlock-free NoC routing algorithm, named *Modulo XY* [16], packets advance in the NoC fabric at a rate of one packet per time-slot [16]. The speedup of a UDN module is defined as the speed ratio at which the fabric can run with respect to the input/output ports. It is equivalent to the fabric removing up to SP packets from one input and send up to SP packets to one output per time slot. On-chip routers make local decisions about the packets next destinations using Round-Robin (RR) arbitration. The UDN switch can sustain high throughput and

low delays under heavy loads if the fabric is running with a small speedup ($SP > 1$). Given the small sized on-chip routers and short wires, a speedup of 2 can be readily affordable. Section IV further studies this property.

C. IM Matching and CM Dispatching

The need for a conflict-free matching in conventional Clos-network switches, such as MSM, mandates the need of two types of matchings [8] [10] [11]: a matching within each input module to select eligible VOQs among non-empty candidate VOQs and a second matching between IMs and CMs. Both of these matchings are quite complex and time consuming due to the high number of input queues per IM for the first matching as well as the global synchronisation of input-output port pairs for IM-CM matching to produce a conflict-free match.

The proposed Clos-UDN greatly simplifies this process as follows. First, each IM needs to maintain only m input queues and each input port of an input module can send to only one FIFO queue per time-slot, making the FIFO running at only twice the line rate. The adoption of NoC fabrics at the central modules of the Clos-UDN obviates the need for maintaining a per-output queue in each input port (VOQs). This is because the NoC is a multi-hop network and packets are routed autonomously in the NoC based on their output destinations encoded in their headers. This makes a FIFO structure sufficient as described in [12]. In the Clos-UDN switch, there are m schedulers in every IM, one per FIFO queue. The RR input schedulers are initialized to different values and they keep updating their selection pointers to one position at the end of every time slot. This guarantees that all pointers are always desynchronized and no conflict in the LI links happens. At the start of every time-slot, a scheduler selects an $LI(i, r)$ link among m links in a RR fashion to transfer the HoL packet from each non-empty FIFO to a central stage/module of the Clos-UDN network. A packet is accepted to the CM module if the left-most NoC router still has room in its left buffer. Once at the NoC, the *Modulo XY* routing algorithm takes over and routes the packet to its outgoing port.

Unlike other types of Clos network switches with bufferless middle stages, our architecture does not require an IM-CM matching. The central modules, NoC fabrics, make parallel and distributed forwarding decisions independently. Routers of the UDN decide about the next hop of transferred packets. They examine and modify the route information continuously until the packet reaches its destination. Packets contending for a link would remain stored to the router's buffers before the arbiter grants them access [12] [15]. Correspondingly, contention for $LC(r, j)$ links gets resolved within the UDN units as packets progress in the NoC as 3 shows. Hence, the process of path-allocation in the Clos network is relaxed and no centralized and global decision and synchronisation are needed.

IV. SIMULATION RESULTS

This section presents the experimental results of the three-stage Clos-UDN with dynamic dispatching scheme. We compare the performance of the Clos-UDN to the MSM switch architecture with the Concurrent Dispatching (CD) [8] and CRRD dispatching as described in [10]. Unless otherwise specified, the default settings of the Clos-UDN switch are such

that the NoC routers buffers depth ($BD=4$), each NoC size is $k \times k$, a full mesh depth is used ($M=k$) and different speedup (SP) values are considered. It is important to note that the speedup (SP) used in the Clos-UDN is different than the conventional speedup [20], where SP refers to the internal switch over-speed factor with respect to the external line rate. Here SP refers to the over-speed factor of *only* the NoC routers inside each UDN central module, excluding the LIs and LCs. Meaning, just like the MSM, the Clos-UDN always sends at maximum one packet per LI/LC link per time slot. Since the Clos-UDN does not use iterations (i.e. time) in its matching, this could be compensated by internally running the UDN CMs with small SP values.

A. Clos-UDN vs. MSM

1) *Bernoulli uniform traffic*: We compare the performance of the Clos-UDN network switch to the MSM switch architecture with the Concurrent Dispatching (CD) [8] and CRRD dispatching as described in [10]. Fig. 4 compares the packets

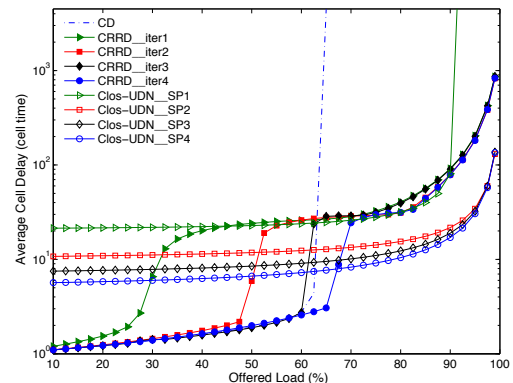


Fig. 4: Delay performance of Clos-UDN and MSM using CD and CRRD algorithms, switch size 64×64 under Bernoulli uniform traffic

delay performance of the Clos-UDN using different speedup values (SP) to that of a MSM switch employing CD and CRRD with different iterations. As shown, Clos-UDN provides higher throughput than MSM with CD dispatching. The Clos-UDN CM units running with a speedup factor of one make the switch achieve 90% throughput. It is the packets progressing in the central switching units by one at each cycle ($SP = 1$) that prevents the switch from achieving full throughput. Increasing the speedup factor to two suffices for the switch to achieve full-throughput. The proposed switch architecture outperforms the MSM with CD and CRRD dispatching schemes with medium-to-high uniform arrivals (which are more relevant). The slightly higher delay experienced by the Clos-UDN under light loads is due to the time required to fill-in the pipeline of the multihop NoC based CMs as. However, Clos-UDN maintains low and almost constant delay irrespective of the traffic load. When the load is larger than 0.7, the delay performance of Clos-UDN with $SP = 2$ becomes better than CRRD with 4 iterations. Clearly, increasing the speedup of UDN switches pulls down the initial delay (for input load less than 30%). Our simulations, as presented in Fig. 5, show that the Clos-UDN switch is robust to switch size variation. It takes advantage of the pipelined nature of the CM units to continuously absorb

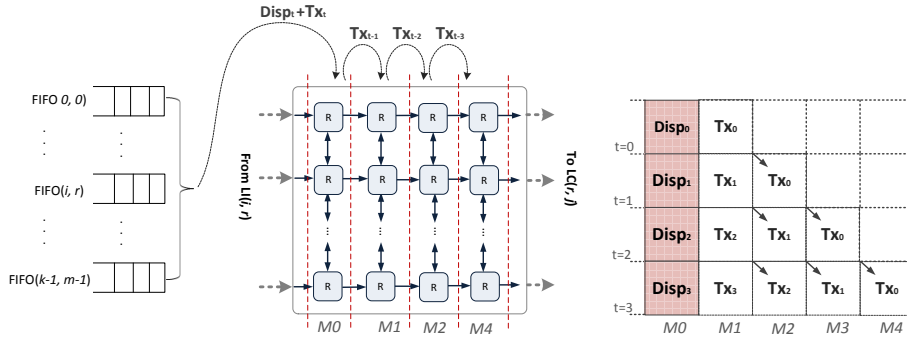


Fig. 3: Pipelined working of Clos-UDN dispatching and packets forwarding through UDN modules.

contention while forwarding packets. Even for a 256×256 switch, the average packets delay is still smooth.

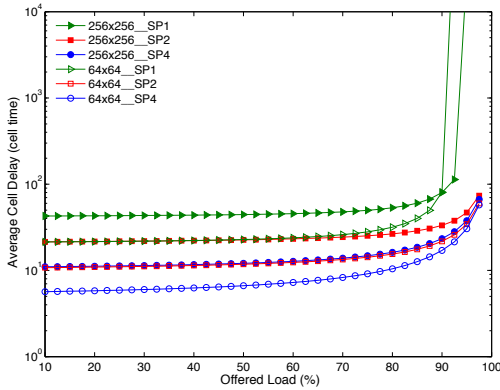


Fig. 5: Performance of Clos-UDN for different switch sizes(size=*, Bernoulli Uniform, $BD = 4$, $SP = *$, $M = full$)

2) *Bursty uniform traffic*: We examine the effect of burstiness on the Clos-UDN switch by considering a bursty traffic with a default burst length equal to 10. Fig. 6 reveals that the delay's growth of the Clos-UDN under bursty arrivals is smoother than that of the MSM with CRRD even if the matching procedure runs 4 iterations. Increasing the number of iterations for the CRRD provides better matching between IMs and CMs and resolves faster the contention which lead to improved switch performance when the load is below 0.7. Increasing the SP reduces the initial delays for the Clos-UDN. Still, a minimum SP of 2 makes the Clos-UDN switch outperform the MSM with CRRD.

3) *Unbalanced traffic*: We evaluate the Clos-UDN switch under non-uniform unbalanced traffic, as specified in [10]. This traffic pattern has one fraction of the total load generated uniformly and the other fraction destined to the output with the same index as the issuing input. If $\omega=0$, then the traffic is perfectly uniform. If $\omega = 1$, the switch deals with a totally unbalanced traffic. Fig. 7 depicts the switch throughput when we vary the unbalancing coefficient ω . The Clos-UDN with $SP = 1$ achieves 90% throughput for $\omega = 0$ (uniform traffic), as has been already shown in Fig. 4. The switch throughput increases with increasing ω . As shown, the Clos-UDN is more

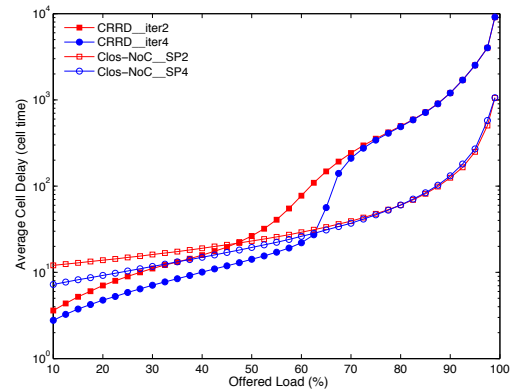


Fig. 6: Delay performance of Clos-UDN and MSM using CRRD, switch size 64×64 under Bursty uniform traffic

stable unlike MSM that reaches as low as 60% throughput using CRRD with 4 iterations and $\omega = 0.5$.

Setting $\omega = 0.5$ corresponds to a non-uniform hot-spot traffic, where 50% of the input load goes to one output while the rest is equally distributed over the remaining outputs. Fig. 8 presents the results, under these settings, for a 64×64 switch operating both the Clos-UDN and MSM architectures. As depicted in Fig. 8, the Clos-UDN switch architecture has much better average delay than the MSM, irrespective of the Clos-UDN speedup and the CRRD number of iterations.

In conclusion, we can observe from the above simulation results that the Clos-UDN running at a speedup of just two is sufficient to have far superior performance than the MSM with up to 4 iterations irrespective of the switch size and/or the traffic settings. Given the NoC routers nature, small on chip with short wires, we conjecture that running them with speedup of two is quite straightforward using current technology and is clearly a better choice than running a centralized and complex scheduling, such as CRRD, for 4 iterations. This makes the Clos-UDN an undoubtedly good alternative for large-scale, high-performance switching architectures.

V. CONCLUSION

This paper proposed a novel multistage switching architecture for DCNs. The Clos-UDN is highly-scalable and easily

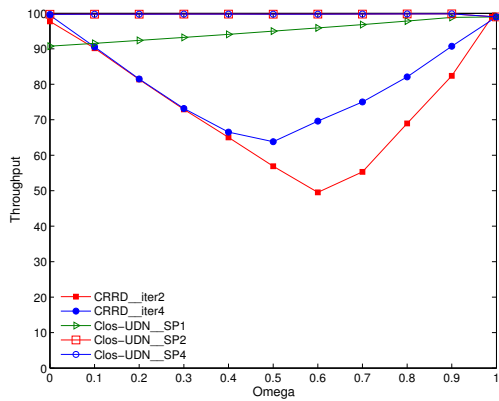


Fig. 7: Throughput stability of Clos-UDN vs MSM under Unbalanced traffic, varying ω

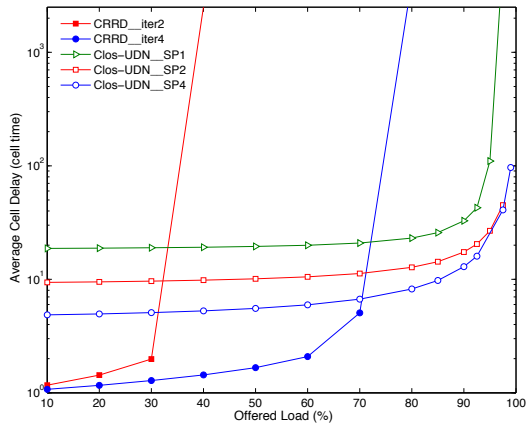


Fig. 8: Delay Performance of Clos-UDN under Unbalanced traffic, $\omega = 0.5$ ($BD = 4$, $SP = 2$, $M = 2$)

configurable. Simple FIFO queuing is used at the input modules along with simple RR dispatching scheme. NoC-based fabrics with distributed on-chip buffering and arbitration units are plugged to the middle stage. The performance of the proposed switching architecture is tested under both uniform and unbalanced traffic models. Simulations showed that the three-stage Clos-UDN provides high throughput and total packets latency better than CRRD for MSM switch. The switch achieves good throughput under unbalanced traffic without any complex dispatching algorithms. The three-stage Clos-UDN is shown to outperform the MSM architecture under all scenarios with NoC modules running a speedup of two.

The Clos-UDN switch with dynamic dispatching process mis-sequences packets delivery. We propose preventing packets from getting dis-ordered at first place by introducing a static configuration of the IM and CM modules interconnections. This results in constantly dispatching packet flows to the same CMs where they are forwarded in a multi-hop way using deterministic routes. The switch architecture is reduced to two-stage Clos network that guarantees in-order packets transfer. A detailed study of the static dispatching scheme and the switch performance is reserved for future work.

REFERENCES

- [1] N. I. Chrysos, "Request-Grant scheduling for congestion elimination in multi-stage networks," Crete University, 2006, Tech. Rep.
- [2] N. Chrysos and M. Katevenis, "Scheduling in non-blocking buffered three-stage switching fabrics," in *INFOCOM*, vol. 6, 2006, pp. 1–13.
- [3] H. J. C. Yu Xia, "On practical stable packet scheduling for bufferless three-stage Clos-network switches," in *HPSR, 2013*. IEEE, 2013, pp. 7–14.
- [4] C. Clos, "A study of non-blocking switching networks," *Bell System Technical Journal*, vol. 32, no. 2, pp. 406–424, 1953.
- [5] H. J. Chao, J. Park, S. Artan, S. Jiang, and G. Zhang, "TrueWay: a highly scalable multi-plane multi-stage buffered packet switch," in *HPSR, 2005*. IEEE, 2005, pp. 246–253.
- [6] E. Oki, N. Kitsuwon, and R. Rojas-Cessa, "Analysis of Space-Space Clos-network packet switch," in *ICCCN 2009*. IEEE, 2009, pp. 1–6.
- [7] Z. Dong, R. Rojas-Cessa, and E. Oki, "Memory-Memory-Memory Clos-network packet switches with in-sequence service," in *HPSR, 2011*. IEEE, 2011, pp. 121–125.
- [8] F. M. Chiussi, J. G. Kneuer, and V. P. Kumar, "Low-cost scalable switching solutions for broadband networking: the ATLANTA architecture and chipset," *IEEE*, vol. 35, no. 12, pp. 44–53, 1997.
- [9] X. Li, Z. Zhou, and M. Hamdi, "Space-Memory-Memory architecture for Clos-network packet switches," in *ICC 2005*. IEEE, 2005, pp. 1031–1035.
- [10] E. Oki, Z. Jing, R. Rojas-Cessa, and H. J. Chao, "Concurrent round-robin-based dispatching schemes for Clos-network switches," *IEEE/ACM*, vol. 10, no. 6, pp. 830–844, 2002.
- [11] J. Kleban and A. Wiczeorek, "CRRD-OG: A packet dispatching algorithm with open grants for three-stage buffered Clos-network switches," in *HPSR, 2006*. IEEE, 2006, pp. 6–pp.
- [12] K. Goossens, L. Mhamdi, and I. V. Senin, "Internet-router buffered crossbars based on networks on chip," in *DSD'09*. IEEE, 2009, pp. 365–374.
- [13] E. Bastos, E. Carara, D. Pigatto, N. Calazans, and F. Moraes, "MOTIM-A scalable architecture for Ethernet switches," in *VLSI, 2007. ISVLSI'07*. IEEE, 2007, pp. 451–452.
- [14] F. Moraes, N. Calazans, A. Mello, L. Möller, and L. Ost, "HERMES: an infrastructure for low area overhead packet-switching networks on chip," *INTEGRATION, the VLSI journal*, vol. 38, no. 1, pp. 69–93, 2004.
- [15] T. Karadeniz, L. Mhamdi, K. Goossens, and J. Garcia-Luna-Aceves, "Hardware design and implementation of a Network-on-Chip based load balancing switch fabric," in *ReConFig, 2012*, pp. 1–7.
- [16] L. Mhamdi, K. Goossens, and I. V. Senin, "Buffered crossbar fabrics based on networks on chip," in *CNSR, 2010*, pp. 74–79.
- [17] A. Bitar, J. Cassidy, N. E. Jerger, and V. Betz, "Efficient and programmable Ethernet switching with a NoC-enhanced FPGA," in *Proceedings of the 10th ACM/IEEE ANCS*. ACM, 2014, pp. 89–100.
- [18] Z. Dong and R. Rojas-Cessa, "Non-blocking Memory-Memory-Memory Clos-network packet switch," in *Sarnoff Symposium, 2011 34th IEEE*. IEEE, 2011, pp. 1–5.
- [19] K. Goossens, J. Dielissen, and A. Radulescu, "Æthereal network on chip: concepts, architectures, and implementations," *Design & Test of Computers, IEEE*, vol. 22, no. 5, pp. 414–421, 2005.
- [20] S.-T. Chuang, A. Goel, N. McKeown, and B. Prabhakar, "Matching output queuing with a combined input/output-queued switch," *Selected Areas in Communications, IEEE Journal on*, vol. 17, no. 6, pp. 1030–1039, 1999.
- [21] L. Mhamdi, "Pbc: A partially buffered crossbar packet switch," *Computers, IEEE Transactions on*, vol. 58, no. 11, pp. 1568–1581, 2009.