Traffic Tolerance of Nanosecond Scheduling on Optical Circuit Switched Data Center Network

Joshua L. Benjamin, Alessandro Ottino, Christopher W. F. Parsonson, Georgios Zervas

University College London, Torrington Place, Bloomsbury, London WC1E 7HB

joshua.benjamin.09@ucl.ac.uk

Abstract: PULSE's ns-speed NP-hard network scheduler delivers skew-tolerant performance at 90% input loads. It achieves >90% throughput, 1.5-1.9 μs mean and 16-24 μs tail latency (99%) for up to 6:1 hot:cold skewed traffic in OCS DCN.

1. Introduction

Data centres handle diverse applications/workloads and need to accommodate a pool of heterogeneous technologies (xPUs,memory,storage). Bandwidth scaling with deterministic throughput and ultra-low latency while minimising power and cost are the key pressing requirements of today's data centre networks (DCNs). In the past decades, optical switches have not been able to replace electronic packet switches in DCNs due to many challenges. Research on optical packet switching (OPS) technology has taken measures to tackle these challenges: the lack of optical buffers/requirement for re-transmission, the need for additional header processing, the complex control required and the competence/compatibility required to match current electronic network technology [1]. However, optical circuit switching (OCS) technology provides guaranteed contention-free connections with a simplified requirement of simpler control decisions and synchronicity. While traditional ms-speed OCS have been purposed to work along with electronic packet switches (HELIOS and c-Through) [2], more recent RotorNet [3] and Sirius [4] have employed a schedule-less approach to benefit from the μs to ns-second speed reconfiguration and low-complexity control plane. However, such scenarios may not efficiently cater for the dynamicity, diversity and skewness of data center network traffic, since demands have to be matched for the scheduling resources (rather than scheduling matching the demand). In PULSE [5], we proposed (i) fast tunable transceivers for sub-ns switching and (ii) an ASIC-based Network schedule Processing Unit (NsPU) for high-throughput low-latency scheduling. In this work, using the custom traffic generator, TrafPy [6], we evaluate the performance of NsPU under skewed traffic (up to 6:1 hot:cold node ratio) and contrast with EPS to highlight the benefits and importance of achieving high skew-tolerant performance. The results show a tolerant scheduling performance with standard deviation, average, median and tail latency of 4.5-7.5 μs, 1.5-1.9 μs, 0.5-0.6 μs and 92-136 μs respectively across all skewed scenarios evaluated at 90% input load. It delivers >90% normalized throughput (in all cases) even at 90% input load, which outperforms all electronic packet switched DCN with relatively deterministic latency.

2. PULSE: OCS Network Architecture

In earlier work [5], the performance of PULSE's NsPU was evaluated under uniform network traffic. To account for emerging applications and heterogeneity of nodes, the PULSE NsPU must be performance-tolerant to traffic demands with various skewness. Hence, in this work, we test the NsPU performance under different skewed traffic flows using a Python-based traffic generation tool, TrafPy [6], to showcase the scheduling throughput and packet latency tolerance under various network loads.

2.1. Architecture

The all-to-all PULSE architecture is composed of passive star-coupler based sub-nets in the core. PULSE effectively enables node-pair communication by selecting the TX-RX (i) wavelength (WDM) using fast tunable (<1 ns) hybrid laser source based on two DS-DBRs and two SOAs for the transmitter [5, 7] and wavelength tunable filter/coherent receiver, (ii) space (SDM) with SOA gates (also compensates for the added optical loss) at the source and destination, (iii) timeslot (TDM) by managing communication in epoch and timeslot numbers. As shown in Fig. 1, in the PULSE OCS architecture, each of the N blades (server node or xPU) in a rack is equipped with x transceivers and each transceiver connects a source cluster to a unique destination cluster. Each transceiver is connected to a $1:p$ splitter, where each path (selection enabled by SOA gates) connects to one of p receiver racks per cluster. As summarised in Table. 1, PULSE achieves 61 pJ/bit/path, while scaling up to 131,072 nodes (N=p=64, x=32), where each node is equipped with a 32 $\times$400 Gbps transceivers i.e. 12.8 Tbps capacity. As compared with

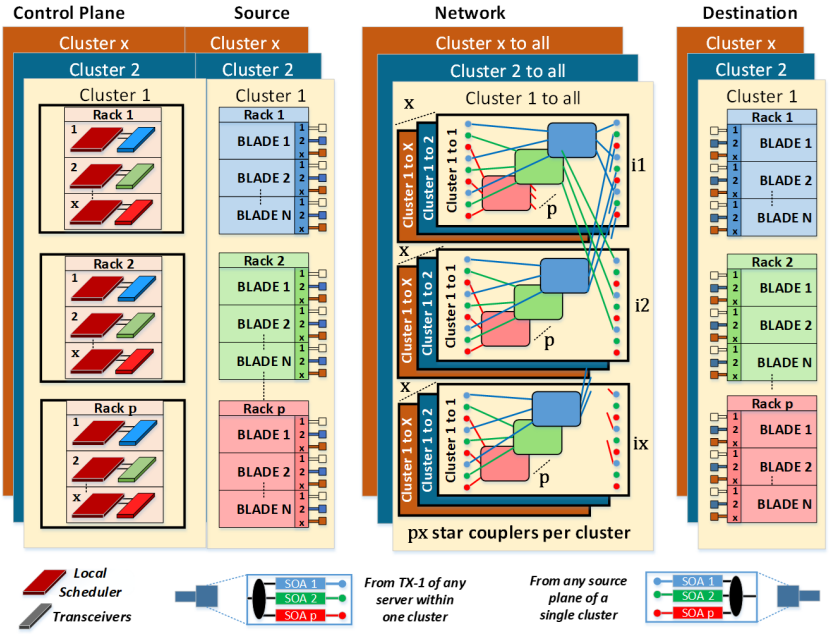

Fig. 1. PULSE Optical Circuit Switched Network Architecture

		EPS DCNs [8]	PULSE
Throughput		-	90%
Max. Net. Utilisation		30-60%	90%
Latency (µs)	Average	58-442	1.5-1.9
	Median	57-447	0.5-0.6
	Tail (99%)	76-1440	16-24
	Max.	268-14360	92-136
	Std. Dev.	6-278	4.5-7.5
BW/node (Gbps)		400	12,800
Energy [5] (pJ/bit)		350	61

Table 1. PULSE achievements: Skew tolerant performance

conventional EPS DCN [8], the throughput and network utilization achieved is relatively higher (90%), while the latency achieved is much lower and tighter in range.

2.2. *Network schedule Processing Unit (NsPU)*

PULSE's NsPU uses contention-resolving arbiters schedule to provide scheduling performance. At the input, the scheduler ingests the source, destination and timeslot size requests, while the outputs are wavelength, timeslot and notifications of grant. As parallelism introduces contention, contention is resolved between sources-destination node pairs every iteration. Successful requests are then qualified to compete for wavelengths and timeslots, which are granted based on a round-robin parallel schedule. In [5], we showed that PULSE NsPU can scale and achieve a clock speed of 435 MHz for a N=64-port sub-network, when synthesized on a 45 nm CMOS library. During timeslot arbitration, p SOA gated paths contend in the NsPU and a parallel round-robin schedule is used to allocate the right path.

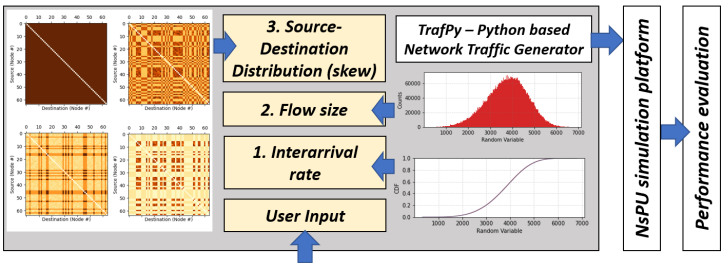

Fig. 2. Evaluation of NsPU performance with TrafPy traffic generator: emulation of custom DCN skew patterns

3. Performance Evaluation and Results

3.1. *Simulation Setup*

Figure 2 shows a highly flexible Python-based traffic generator and data center network benchmarker tool, TrafPy [6]. While the tool is able to generate unique traffic patterns similar to diverse types of cloud services, we used the custom tuning functionality to emulate extreme skew scenarios to characterize the performance tolerance of PULSE NsPU. The NsPU was run on TrafPy generated traces for 5 seeds with 5000 epochs each to demonstrate confidence intervals above 95%. The input skews are shown by the source-destination request size map in the inset in Fig. 2 with 0 (uniform), 16, 32 and 48 skewed nodes per 64 node sub-net. The skew in traffic is defined as the traffic requested by skewed (hot or cold) nodes divided by non-skewed nodes. It is evident that extreme skew scenarios are mostly found at low and medium loads and less so on very high loads, where skew is shown by 3(b) (second Y-axis).

3.2. *NsPU: A traffic skew tolerant performance*

As shown in Fig. 3(a), for all skewed traffic the sustained throughput is always above 90% of the input network load; at the heaviest 90% input network load, a throughput of above 81% (normalized throughput >90%) is

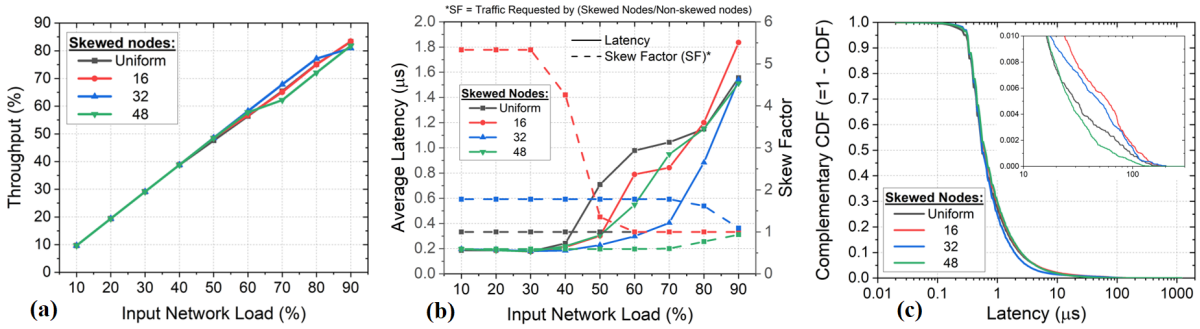

Fig. 3. PULSE NsPU Skew Tolerance: (a) Sustained normalized throughput >90%, (b) Average latency of $<2\mu$s at various input network loads, (c) Tail latency below $<200\mu$s at 90% input load.

achieved for 0, 16. 32 and 48 skewed nodes (out of 64 nodes per sub-net). Here, we see that even with increased divergence away from traffic uniformity, still a sustained throughput is achieved. This high tolerance and resilience of the NsPU to achieve consistent performance is attributed to how the scheduling is performed. The scheduler, in parallel, cater to long-lived requests from the buffer with coarse allocation (multiple slots per iteration) and provide fine allocation to the remaining non-contending grants.

In Fig. 3(b), the average packet latency for various input network loads is shown by the solid lines, while the skewness is indicated by the dashed lines. Uniform traffic scheduling sees an increase in latency above 40% network load, when contention and competition for grants start to increase. A similar effect is seen for networks with 16 and 48 skewed nodes as well, while 32 skewed nodes (50%) nodes skewed sees a minimal increase in latency as contention decreases. However, at a heavy load of 90%, the latency is also between 1.5-1.9μs; the worst-case scenario occurs only when skewed nodes is 16 (25%) and at increased loads, where there is some room for tolerance improvement. In Fig. 3(c), the complementary CDF of all scheduled packets are shown to have a worst-case median latency of 0.6μs, tail latency (99th percentile) of 24μs, and a maximum of 136μs in the zoomed in inset. As expected, the highest tail latency is in the 16 skewed nodes scenario. The performance improvements achieved over conventional EPS DCNs is shown numerically in Table 1.

Whilst we have shown that this hardware based scheduling heuristic can achieve high performance tolerance for diverse dynamic traffic patterns, we are now in the process of developing flexible and programmable features with high level control and monitoring facilities for users.

4. Conclusion

A scalable all optical circuit switched DCN architecture PULSE with packet timescale circuit reconfiguration processor (NsPU) was reported. We showcased that the control and data plane can be arranged in modular structures called sub-nets up to x clusters, scaling up to 131,072 nodes with 32 transceivers per node, 12.8 Tbps capacity per node, achieving >92 % normalised sustainable throughput, a low median ($<0.6\mu$s) and tail latency ($<24\mu$s) at 240 ns epoch. The gain-gate broadcast and select modules as part of the transceivers, allocation and scheduling strategies that enabled highly tolerant performance under dynamic traffic skew was shown. The achievements of PULSE and improvements offered over EPS multi-level networks in terms of energy (6x less) and the opportunity created for an all-to-all bandwidth environment on adoption of PULSE is exhibited with ultra-low latency.

References

1. N. Calabretta and W. Miao. *Optical switching in data centers: architectures based on optical packet/burst switching*, pages 45–69. Springer, Germany, August 2017.
2. C. Kachris *et al.* A Survey on Optical Interconnects for Data Centers. *IEEE Communications Surveys Tutorials*, 14(4):1021–1036, 2012.
3. W. Mellette *et al.* RotorNet: A Scalable, Low-complexity, Optical Datacenter Network. *SIGCOMM*, pages 267–280, 2017.
4. H. Ballani *et al.* Sirius: A Flat DCN with Nanosecond Optical Switching. ACM SIGCOMM, August 2020.
5. J. L. Benjamin *et al.* PULSE: OCS Data Center Architecture Operating at Nanosecond Timescales. *JLT*, 38(18):4906–4921, Sep 2020.
6. C. Parsonson *et. al.* Trafpy: Benchmarking data centre network systems. *arXiv:2107.01398*, 2021.
7. T. Gerard *et. al.* Ai-optimised tuneable sources for bandwidth-scalable, sub-nanosecond wavelength switching. *Opt. Express*, 29(7):11221–11242, Mar 2021.
8. University of Cambridge. Latency-driven performance in data centres, 2019. Accessed: October 19, 2021 [Available Online]: https://www.cl.cam.ac.uk/techreports/UCAM-CL-TR-937.pdf.