

Cycle-accurate evaluation of reconfigurable photonic networks-on-chip

Christof Debaes,^a Iñigo Artundo,^b Wim Heirman,^c
Jan Van Campenhout^c and Hugo Thienpont^a

^aVrije Universiteit Brussel, Belgium;

^bUniversidad Politécnica de Valencia, Spain;

^cGhent University, Belgium

ABSTRACT

There is little doubt that the most important limiting factors of the performance of next-generation Chip Multiprocessors (CMPs) will be the power efficiency and the available communication speed between cores. Photonic Networks-on-Chip (NoCs) have been suggested as a viable route to relieve the off- and on-chip interconnection bottleneck. Low-loss integrated optical waveguides can transport very high-speed data signals over longer distances as compared to on-chip electrical signaling. In addition, with the development of silicon microrings, photonic switches can be integrated to route signals in a data-transparent way. Although several photonic NoC proposals exist, their use is often limited to the communication of large data messages due to a relatively long set-up time of the photonic channels. In this work, we evaluate a reconfigurable photonic NoC in which the topology is adapted automatically (on a microsecond scale) to the evolving traffic situation by use of silicon microrings. To evaluate this system's performance, the proposed architecture has been implemented in a detailed full-system cycle-accurate simulator which is capable of generating realistic workloads and traffic patterns. In addition, a model was developed to estimate the power consumption of the full interconnection network which was compared with other photonic and electrical NoC solutions. We find that our proposed network architecture significantly lowers the average memory access latency (35% reduction) while only generating a modest increase in power consumption (20%), compared to a conventional concentrated mesh electrical signaling approach. When comparing our solution to high-speed circuit-switched photonic NoCs, long photonic channel set-up times can be tolerated which makes our approach directly applicable to current shared-memory CMPs.

Keywords: Network-on-Chip, Optical interconnects, Network reconfiguration, Silicon microrings, Power consumption.

1. INTRODUCTION

Power efficiency has become one of the prime design consideration within today's ICT landscape. As a result, power density limitations at the chip level have placed constraints on further clock speed improvements and pushed the field into increased parallelism. This has led to the development of multicore architectures or chip multiprocessors (CMPs). As such, CMPs have begun to essentially resemble highly parallel computing systems integrated on a single chip. One of the most promising paradigm shifts that has emerged in this domain are packet-switched networks-on-chips (NoCs).¹ Since interconnect resources in these networks are shared between different data flows, they can operate at significantly higher power efficiencies than fixed interconnect topologies. However, due to the relentless increase in required throughput and number of cores, the links of those networks are starting to stretch the capabilities of electrical wires. In fact, some recent CMP prototypes with eighty cores show that the power dissipated by the NoC accounts for up to 25 percent of the overall power.²

Meanwhile, recent developments in integrating photonic devices within CMOS technology have demonstrated photonic interconnects as a viable alternative for high performance off-chip and global on-chip communication.³ This has sparked interest among several research groups to propose architectures with photonic NoCs.⁴⁻⁶ However, using optical links as mere drop-in replacements for the connections of electronic packet-switched networks is not the end. Conversion at each routing point from the optical to the electrical domain and back can be power inefficient and increase latency. But novel components, such as silicon microring resonators,⁷ which can now be integrated on-chip, are opening new possibilities to build optical, switched interconnection networks.^{8,9}

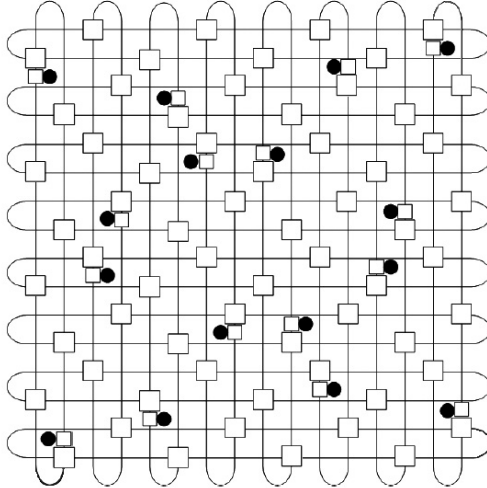


Figure 1. 16-node non-blocking torus.¹⁴ Squares represent optical routers based on microring resonators, and network nodes are represented by discs. The electrical control (or base) network, which is a 2-D torus overlaid on the optical network, is not shown here for clarity.

Lacking a cheap and effective way of optically controlling the routing (and doing possible buffering), these approaches necessarily work in a circuit-switched way. And while the actual switching of the optical components can nowadays be done in mere nanoseconds or less,¹⁰ the set-up of an optical circuit still requires at least one network round-trip time, which accounts for several tens of nanoseconds. This makes that such proposals only reach their full potential at large packet sizes, or in settings where software-controlled circuit switching can be used with relatively long circuit lifetimes. Indeed, in Shacham’s proposal,⁸ packets of several kilobytes are needed to reach a point where the overhead of setting up and tearing down the optical circuits (which is done with control packets sent over an electrical network), can be amortized by the faster optical transmission.

In SoC architectures, and to a lesser extent in CMPs, large direct memory access (DMA) transfers can reach packet sizes of multiple KiB. However, most packets are coherence control messages and cache line transfers. These are usually latency bound and very short. In practice, this would mean that most of the traffic would not be able to use the optical network, as they do not reach the necessary size to compensate for the latency overhead introduced, and that the promised power savings could not be realized!*

We propose to use the combination of the electrical control network and the optical circuit-switched links as a packet-switched network with ‘slow reconfiguration.’ This idea is based on existing work such as the Interconnection Cached Network¹¹ (or see Barker et al.¹² for a modern application). But rather than relying on application control of the network reconfiguration, which requires explicit software intervention and does not agree with the implicit communication paradigm of the shared memory programming model, our approach provides for an automatic reconfiguration based on the current network traffic. This concept has been described by Artundo et al.,¹³ and was proven to provide significant performance benefits in (off-chip) multiprocessor settings. In this paper, we will extend this approach to on-chip networks, trying to model an architecture close to the one introduced by Shacham⁸ and Petracca.¹⁴

2. ON-CHIP PHOTONIC INTERCONNECT ARCHITECTURE

The photonic NoC proposed by Petracca et al.¹⁴ introduces a non-blocking torus topology, connecting the different cores of the system, based on a hybrid approach: a high-bandwidth circuit-switched photonic network combined with a low-bandwidth packet-switched electronic network. This way, large data packets are routed

*One might consider using a larger cache line size to counter this, but an increase to multiple KiB would in most cases only result in excessive amounts of *false sharing*, negating any obtained performance increase.

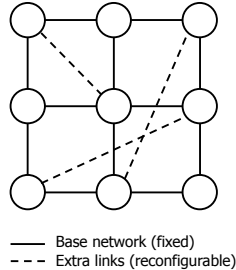


Figure 2. Reconfigurable network topology. The network consists of a base network (a 2-D torus in our architecture), augmented with a limited number of direct, reconfigurable links (which are made up of the reconfigurable optical layer from Figure 1).

through a time and wavelength multiplexed network, for a combined bandwidth of 960 Gb/s, while delay-critical control packets and data messages with sizes below a certain threshold are routed through the low-latency electrical layer. As the basic switching element, a 4×4 hitless silicon router is presented by Sherwood-Droz et al.,¹⁵ based on eight silicon microring resonators with a bandwidth per port of 38.5 GHz on a single wavelength configuration.

An example 16-node architecture is depicted in Fig. 1. Each square represents a 4×4 router containing eight microring resonators. In this architecture, each node has a dedicated 3×3 router to inject and eject packets from the network, represented by the smaller squares. The network nodes themselves are represented by discs. By means of the electronic control layer, each node first sends a control packet to make the reservation of a photonic circuit from source to destination. Once this is done, transmission is done uninterrupted for all data packets. To end the transmission phase, a control packet is sent back from the destination to free the allocated resources.

For our architecture, a dedicated reconfigurable photonic layer will be used as a data transmission layer, where a set of extra links will be established in a circuit-switched fashion for certain intervals of time, depending on automated load measurements over the base topology. The reconfiguration will follow slow-changing dynamics of the traffic, while the base electronic network layer will still be there to route control and data messages.

Other similar architectures have been proposed, such as Gu's,¹⁶ where the need for an electrical control layer has been removed and all packets are sent through an all-optical network using different wavelengths. Still, the separation between control and data layers, even when they are sent through the same physical channels, is maintained. Our approach is valid to any network architecture where this distinction is kept, as the reconfigurable layer can be virtually established irrespective of the underlying physical implementation.

3. RECONFIGURABLE OPTICAL LAYER

3.1 Using traffic locality to trigger reconfiguration

It is known that memory references exhibit locality in space and time, in a fractal or self-similar way.¹⁷ This locality is exploited by caches. Due to the self-similar nature of locality, this effect is present at all time scales, from the very fast nanosecond scales exploited by first-level caches, down to micro- and millisecond scales which are visible on the interconnection network of a shared-memory (on-chip or multi-chip) multiprocessor. This behavior can be modeled as traffic bursts: these are periods of high-intensity communication between specific processor pairs. These bursts were observed^{13, 18} to be active for up to several milliseconds, on a background of more uniform traffic with a much lower intensity.

From this observation came the idea to use slowly reconfigurable but high (data-) speed optical components to establish temporary 'extra links,' providing direct connections between pairs of processor cores that are involved in a communication burst. Other communication, which is not part of a burst – or a lower-intensity burst when the hardware would support less extra links than there are bursts at a given time – will be routed through a standard packet-switched (optical or electrical) network (the 'base network,' see Figure 2). The positions of the extra links are re-evaluated over time as old bursts stop and new ones appear.

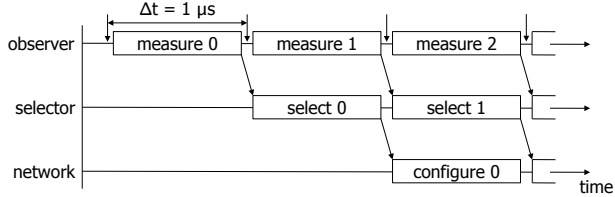


Figure 3. Sequence of events in the on-chip reconfigurable network. During every reconfiguration interval of $1 \mu\text{s}$, traffic patterns are measured. In the next interval, the optimal network configuration is computed for such patterns. One interval later, this configuration is enabled. The reconfiguration itself takes place at the start of each *configure* box, but the switching time is very short (2% of the switching time) in this architecture and is therefore not shown here.

We have previously evaluated this concept in the context of shared-memory servers and supercomputers, and proposed an implementation using low-cost optical components.¹³ Since then, multicore technology has enabled the integration of a complete shared-memory multiprocessor on a single chip. At the same time, on-chip reconfigurable optical interconnects became a reality, using the integration possibilities allowed by the emerging field of silicon photonics.^{19,20}

3.2 Proposed reconfigurable network architecture

Our network architecture, originally proposed by Heirman et al.,²¹ starts from a base network with fixed topology. In addition, we provide a second network that can realize a limited number of connections between arbitrary node pairs – the *extra links* or *elinks*. A schematic overview is given in Figure 2.

The elinks are placed such that most of the traffic has a short path (a low number of intermediate nodes) between source and destination. This way, a large percentage of packets has a correspondingly low (uncongested) latency. In addition, congestion is lowered because heavy traffic is no longer spread out over a large number of intermediate links. For the allocation of the elinks, a heuristic is used that tries to minimize the aggregate hop distance traveled multiplied by the size of each packet sent over the network, under a set of implementation-specific conditions that describe what elinks can be concurrently supported by the architecture. A more detailed description of the underlying algorithms can be found in a study by Heirman et al.,²² the application to Shacham’s NoC architecture using microrings was first described by Artundo.¹⁸

Although the actual reconfiguration, done by switching the microrings, happens in mere picoseconds, the execution time of the optimization algorithm, which includes collecting traffic patterns from all nodes and distributing new configuration and routing data, cannot be assumed negligible. The time this exchange and calculation takes will be denoted by the *selection time* (t_{se}). The actual switching of optical reconfigurable components will then take place during a certain *switching time* (t_{sw}), after which the new set of elinks will be operational. Traffic cannot be flowing through the elinks while they are being reconfigured. Therefore, the reconfiguration process starts by draining all elinks before switching any of the microrings. This takes at most 20 ns (the time to send our largest packet, which is 80 bytes, over a 40 Gbps link). During the whole reconfiguration phase, network packets can still use the base network, making our technique much less costly than some other more intrusive reconfiguration schemes, where all network traffic needs to be stopped and drained from the complete network during reconfiguration.

The reconfiguration interval, denoted by Δt , must be chosen as short as possible to be able to follow the dynamics of the evolving traffic and get a close-to-optimal topology. On the other hand, it must be significantly larger than the switching time of the chosen implementation technology to amortize the fraction of time that the *elinks* are off-line. With optical components that can switch in the 30 ps range, the switching time (t_{sw}) will only take a negligible fraction of the reconfiguration interval Δt . However the selection time (t_{se}) will remain significant as it requires exchange of data over the network.

We therefore propose a scheduling where we allow the selection to take up to a full reconfiguration interval. The three phases (shown in Figure 3) of collecting traffic information (*measure*), making a new elink selection (*select*), and adjusting the network with this selection (*configure*) are performed in a pipelined fashion, where each phase uses the results (traffic pattern or elink selection) of the previous interval.

4. METHODOLOGY

4.1 Simulation platform

We have based our simulation platform on the commercially available Simics simulator.²³ It was configured to simulate a multicore processor inspired by the UltraSPARC T1/T2, which runs multiple threads per core (four in our experiments). This way, the traffic of 64 threads is concentrated on a 16-node network, stressing the interconnection network with aggregated traffic. Stall times for caches and main memory are set to conservative values for CMP settings (2 cycles access time for L1 caches, 19 cycles for L2 and 100 cycles for main memory). Cache coherence is maintained by a directory-based coherence controller at each node, which uses a full bit-vector directory and an MSI protocol. The interconnection network models a packet-switched 4×4 network with contention and cut-through routing. The time required for a packet to traverse a router is 3 cycles. The directory controller and the interconnection network are custom extensions to Simics. Both the coherence traffic (read requests, invalidation messages etc.) and data traffic are sent over the base network. The resulting remote memory access times are around 100 ns, depending on network size and congestion.

The proposed reconfigurable NoC has been configured with a link throughput of 10 Gb/s in the base network. To model the elinks, a number of extra point-to-point links can be added to the base torus topology at the start of each reconfiguration interval. The speed of these reconfigurable optical elinks were assumed to be four times faster than that of the base network links (40 Gb/s). For evaluation, we have compared the proposed solution with three standard, non-reconfigurable NoCs: a 10 Gb/s electrical NoC, a 40 Gb/s electrical NoC and a 40 Gb/s photonic NoC.

The SPLASH-2 benchmark suite²⁴ was chosen as the workload. It consists of a number of scientific and technical algorithms using a multi-threaded, shared-memory programming model (`barnes`, `cholesky`, `fft`, `radix`, `ocean.cont`, `water.sp`). Since the detailed simulation of a single SPLASH-2 benchmark program takes a significant amount of computation time, we used synthetic traffic traces instead. For each of the benchmark applications an individual trace is constructed using the methodology introduced by Heirman.²⁵ This way, we could quickly yet accurately simulate the performance and power consumption of our network under realistic traffic conditions.

4.2 Power measurements

To measure the power consumption of our optical circuit-switched routing, we will need to know the state of each switch in the mesh – this means which microrings are powered on for each reconfiguration interval. We can know this by looking at the routing table of each router (Table 1b in Sherwood et al.¹⁵) and assign a power value for each active ring. In this reference the power consumed per ring in the ON state is assumed to be 6.5 mW, while in the OFF state the required power is considered negligible. This is for rings that switch in only 30 ps, though. Using a reconfiguration interval of one microsecond, our architecture does not need such an exorbitantly fast (and power hungry) device. Instead, it can tolerate several nanoseconds of switching time, and we will assume that such a device can be powered with just 0.5 mW.

Also, Sherwoord et al.¹⁵ consider nine possible states of the router, determined by all possible simultaneous connections between its in- and output ports. Each of these states has a specific number of microrings powered on. However, when a router is only used by a single traversing elink, fewer active microrings are required. If we do not use just the nine predefined states, but only account for the minimal number of rings needed for establishing the optical elink path, we can obtain a significantly lower power consumption.

Therefore, we will assume the use of a more power-efficient scheme that only powers the rings needed on each reconfiguration interval, instead of putting the switch in a state where several rings will be powered whether they are used or not. Of course, the electronic control of such a switch would be more complicated, this is why the nine predefined states were originally proposed even if this is not the most power-efficient scheme. But where the localized control, and the aim for independence between the different circuits validates such an approach, our architecture on the other hand performs a global and simultaneous assignment of all elinks and microrings, and should therefore be able to operate in the optimized case.

For the parameters to estimate the power consumption of the links and the routing of the packets, we have used the same values as cited by Sacham et al.⁸ and shown in Table 1. One notable difference is that we include an extra static power of 500 μ W for each optical link, as it is likely that the analog optical transceiver circuits

Technology Node	32 nm
Core Dimension	$1.67 \times 1.67 \text{ mm}^2$
Electrical Link Power	0.34 pJ/bit/mm
Optical Link Power	0.5 pJ/bit
Buffering Energy	0.12 pJ/bit
Routing Energy	0.35 pJ/bit
Crossbar Transfer Energy	0.36 pJ/bit
Static Electrical Link Power @ 10 Gbps	500 μ W
Static Electrical Link Power @ 40 Gbps	2 mW
Static Optical Link Power	500 μ W
Microring ON Power	500 μ W
Microring OFF Power	0 μ W

Table 1. Power consumption figures.

will consume power even while the links are not sending data. As for the dynamic power dissipated by the E/O and O/E conversion, a reasonable estimate for a modulator and its corresponding detector at 10 Gb/s is 2 pJ/bit. Future predictions push this value down to 0.2 pJ/bit.²⁶ In our simulation we have used a less stringent 0.5 pJ/bit.

5. SIMULATION RESULTS

A direct comparison with our reference architecture¹⁵ is difficult, since in the original case, only large DMA transfers (of which there are usually very few in realistic CMP systems) would use the optical network, while most of the traffic – both by aggregate size and by latency sensitivity – necessarily sticks to the electrical ‘control’ network. Yet, just comparing the performance of our solution with a base-network only architecture is not very insightful either. Therefore, we have made a performance and power comparison of our proposed architecture versus a non-reconfigurable 2-D torus topology.

5.1 Network performance

In this section we first aim to obtain the performance improvement by introducing reconfiguration in the system, versus a standard topology. For this, we compare four approaches: using either the reconfigurable architecture introduced above, or a 2-D torus-only network with link speeds of 10 Gb/s (‘low speed electrical NoC’) or 40 Gbps (‘high speed’) electrical or optical NoC, without reconfiguration capabilities. In the case of an all-optical network, every node needs an optical transceiver in all four directions. Also, a conversion from the optical to the electrical domain is needed at each hop, since the routing is still performed electronically. In contrast, our proposed reconfigurable NoC will require only one transceiver per node, which is an advantage in cost and power consumption. Moreover, the data can now travel over much longer distances until O/E and E/O conversions are needed, which again reduces power and latency.

In Figure 4 and Table 2, average remote memory access latencies are presented for all network configurations. We can observe that the reconfigurable approach performs significantly better than the low-speed non-reconfigurable network (35%), but still far from a high-speed electrical/optical implementation due to the huge amount of bandwidth available in these cases.

In Figure 5 and Table 2, we show the average number of hops per byte sent. Comparing with the non-reconfigurable topology – in which the network consists of just a 2-D torus – we obtain a clear, 22% reduction of the hop distance. Similar simulations on larger scale CMPs with up to 64 cores show a 34.7% reduction in hop distance. This will increase further as the network scales.¹⁸

There is only a small variability between the different applications measured because, at any time, there is exactly the same number of elinks present. The only thing that can differ is that, sometimes, slightly longer routes are created, but since the elink selection always tries to maximize $data \times hop\text{-}distance$, the average hop

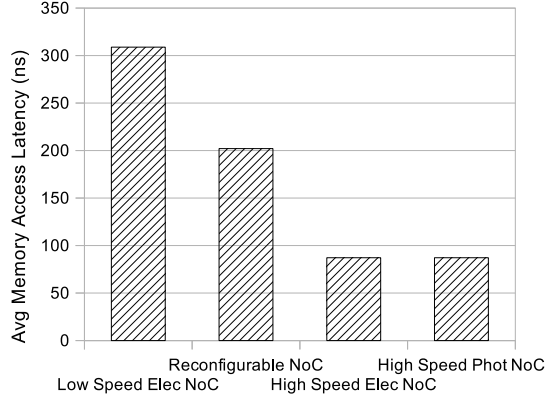


Figure 4. Average remote memory access latency.

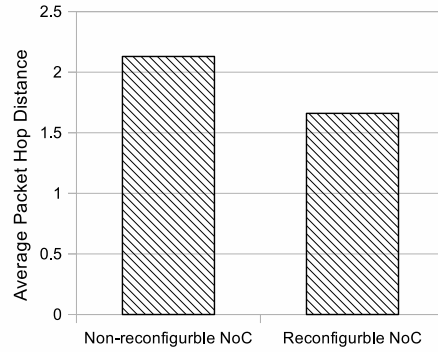


Figure 5. Average number of hops per byte sent.

distance will also be not that different. Note that the number of active microrings depends on the shape of the traffic pattern (the source-destination pair distribution) – albeit not by a great amount – but it does not depend on the traffic magnitude.

5.2 Network usage

A key factor in understanding the power consumption is the usage of the switches and links in the network. For a normal $r \times \frac{p}{r}$ torus topology, the diameter (maximum number of hops between any node pair) is:²⁷

$$D = \left\lfloor \frac{r}{2} \right\rfloor + \left\lfloor \frac{p}{2r} \right\rfloor \quad (1)$$

where p is the number of processors and r is the size of the torus. In regular tori this makes $D = 4$ hops when $p = 16$. For our benchmark applications, the average hop distance is 2.13 for $p = 16$.

In our simulations, we use a folded torus topology as shown in Figure 1. The complete topology contains $\frac{p^2}{4}$ hitless switches (4×4 optical routing elements) and p gateway switches. We found that the mean number of (non-gateway) switches used per elink during each reconfiguration interval is 3.28. This results in a total of 37.5 active optical routing elements (out of the 64 available ones), of which 13 routers are traversed by more than one elink. From all routers, on average 73.7 microrings are in the active state.

Table 2 furthermore details the average data volume over the different NoC architectures. For the proposed reconfigurable NoC we can see that the total volume is almost evenly distributed between the electrical base links and the high-speed reconfigurable elinks. This clearly indicates that the heuristic to allocate the reconfigurable

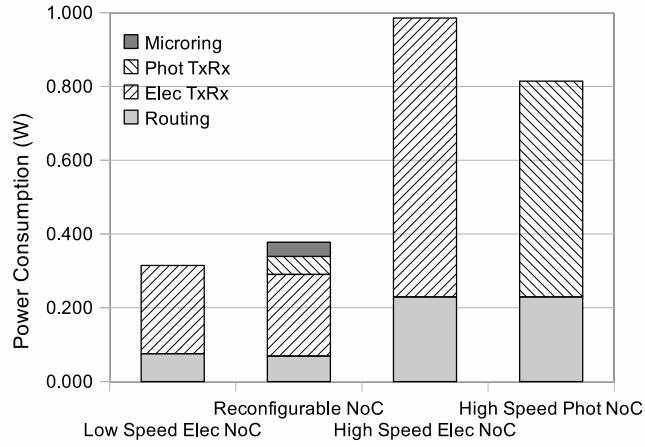


Figure 6. Total power consumption per interval under different network architectures.

	BW_{max} (Gbps)	BW_{avg} (Gbps)	T_{mem} (#cycles)	d_{hop} (#hops)	P_{tot} (mW)
Electrical NoC	10	5.70	308.9	2.13	315
Reconfigurable NoC			202.1	1.66	378
- Base Elec NoC	10	5.21			
- Reconf Phot NoC	40	5.08			
High-speed NoC	40	17.28	87.2	2.13	
- Electrical NoC					985
- Photonic NoC					814

Table 2. Comparison of the link activity and average remote memory access latency for the different types of Networks-on-Chip.

links is able to capture a significant part of data packets in bursts. This figure could nevertheless be further improved when the number of cores and the traffic demands are scaled up.

The folded torus topology used in our study has twice the wire demand and bisection bandwidth of a mesh network, trading a longer average flit transmission distance for fewer routing hops. While wider flits and a folded topology can increase link bandwidth utilization efficiency, this remains still low in our simulations, as shown in Table 2. Pande et al.²⁸ investigated various metrics of a folded torus NoC, including energy dissipation, for different traffic loads. The comparative analysis was done with respect to average dynamic energy dissipated per full packet transfer from source to destination node. It was found that energy dissipation increases linearly with the number of virtual channels (VCs) used. Furthermore, a small number of VCs will keep energy dissipation low without giving up throughput. Energy dissipation reaches an upper limit when throughput is maximized, meaning that energy dissipation does not increase beyond the link saturation point. In general, architectures with more elaborated topologies, and therefore higher degrees of connectivity, have a higher energy dissipation on average at this saturation point than do others. If power dissipation is critical – which is usually the case in on-chip multiprocessor networks – a simpler mesh topology may be preferable to a folded torus, as detailed in the work by Dally et al.¹

5.3 Power consumption

In this section, we evaluate the power consumed by the NoCs and include the powering of the microring resonators when establishing the elinks on the reconfigurable layer.

An estimation of the power consumption consumed by the NoC can be calculated by combining the parameters given in Table 1 and the activity of the links and optical switches in sections 5.1 and 5.2. In comparison to the low-speed NoC with fixed topology, the reconfigurable NoC consumes modestly more power (20%) and improves significantly averaged network performance. Moreover, in comparison to the high-speed fixed NoCs, the proposed solution consumes significantly less (corresponding to a reduction from 54% to 62% as compared to the fixed photonic and electrical NoCs).

It is important to note at this stage that we have adopted rather conservative memory stall times (see section 4.1). Future CMPs, equipped with improved cache hierarchies, will impose significantly higher throughput demands on the intercore network and further increase the power consumption of the NoC. In addition, the proposed solution based on the reconfigurable NoC will benefit from this scaling as it will decrease the network traffic contention between the most active communicating pairs.

The estimated power consumption is of course highly dependent on the parameters chosen in Table 1 which was taken from Sacham et al.⁸ Nevertheless, the conclusions that we draw from the results are generic. The proposed reconfigurable NoCs will always perform better than the fixed NoC consisting solely of a base network. The reason is that in our proposal, links with more bandwidth and lower latency are added only where and when relevant. When compared to high-speed NoCs our proposal consumes less power since it requires much less high-speed links and transceivers. The proposed photonic NoC thus allows for a very efficient resource utilization of the high-speed transceivers.

In our study, we assumed that the silicon microrings do not consume energy in their off-state. This justifies our choice to adopt the proposal by Sacham et al.⁸ for the photonic links, where a network of p nodes requires $8p^2$ microring switches (excluding the gateway switches). Temperature detuning of the microrings might require extra power dissipation to stabilize the temperature locally at each ring. In recent work,²⁹ however, silicon microrings were demonstrated with a temperature dependence as low as 0.006 nm/°C.

6. CONCLUSIONS

We introduced a reconfigurable optical interconnect for a NoC multicore system that makes use of ultra-low power photonic switches to route messages over a reconfigurable optical layer, while keeping an underlying electronic base network. Since we allow for slow reconfiguration or adaptation of the optical layer to the current traffic pattern, our approach can make much better use of the optical layer – which otherwise would only be beneficial for very long packets, or for circuits that were explicitly set up by the programmer. Both these conditions are however not compatible with realistic chip multiprocessor architectures.

By using our approach, however, the full benefits of optical switching can be combined with realistic CMP conditions, paving the way for photonic interconnects to satisfy the future bandwidth needs of large multicore designs.

Acknowledgments

This work was supported by the European Commission's 6th FP Network of Excellence on Micro-Optics (NEMO), the BELSPO IAP P6/10 photonics@be network sponsored by the Belgian Science Policy Office, the GOA, the FWO, the OZR, the Methusalem and Hercules foundations. The work of C. Debaes is supported by the FWO (Fund for Scientific Research - Flanders) under a research fellowship.

REFERENCES

- [1] Dally, W. J. and Towles, B., "Route packets, not wires: On-chip interconnection networks," in [*Design Automation Conference*], 684–689 (June 2001).
- [2] Hoskote, Y., Vangal, S., Singh, A., Borkar, N., and Borkar, S., "A 5-GHz mesh interconnect for a Teraflops processor," *IEEE Micro* **27**, 51–61 (Sept. 2007).
- [3] Owens, J. D., Dally, W. J., Ho, R., Jayasimha, D., Keckler, S. W., and Peh, L.-S., "Research challenges for on-chip interconnection networks," *IEEE Micro* **27**(5), 96–108 (2007).
- [4] Ohashi, K., Nishi, K., Shimizu, T., Nakada, M., Fujikata, J., Ushida, J., Torii, S., Nose, K., Mizuno, M., Yukawa, H., Kinoshita, M., Suzuki, N., Gomyo, A., Ishi, T., Okamoto, D., Furue, K., Ueno, T., Tsuchizawa, T., Watanabe, T., Yamada, K., Itabashi, S., and Akedo, J., "On-chip optical interconnect," *Proceedings of the IEEE* **97**, 1186–1198 (July 2009).
- [5] Brière, M., Girodias, B., Bouchebaba, Y., Nicolescu, G., Mieyeville, F., Gaffiot, F., and O'Connor, I., "System level assessment of an optical NoC in an MPSoC platform," in [*Design, Automation and Test in Europe (DATE)*], 1084–1089 (2007).
- [6] Beausoleil, R. G., Ahn, J., Binkert, N., Davis, A., Fattal, D., Fiorentino, M., Jouppi, N. P., McLaren, M., Santori, C. M., Schreiber, R. S., Spillane, S. M., Vantrease, D., and Xu, Q., "A nanophotonic interconnect for high-performance many-core computation," *IEEE LEOS Newsletter*, 15–22 (June 2008).
- [7] Xu, Q., Fattal, D., and Beausoleil, R. G., "Silicon microring resonators with 1.5- μm radius," *Optics Express* **16**(6), 4309 (2008).
- [8] Shacham, A., Bergman, K., and Carloni, L., "Photonic networks-on-chip for future generations of chip multi-processors," *IEEE Transactions on Computers* **57**, 1246–1260 (Sept. 2008).
- [9] Koochi, S. and Hessabi, S., "Contention-free on-chip routing of optical packets," in [*Proceedings of the 3rd ACM/IEEE International Symposium on Networks-on-Chip*], 134–143 (May 2009).
- [10] Fidaner, O., Demir, H. V., Sabnis, V. A., Zheng, J.-F., Harris, J. S. J., and Miller, D. A. B., "Integrated photonic switches for nanosecond packet-switched optical wavelength conversion," *Optics Express* **14**(1), 361 (2006).
- [11] Gupta, V. and Schenfeld, E., "Performance analysis of a synchronous, circuit-switched interconnection cached network," in [*ICS '94: Proceedings of the 8th international conference on Supercomputing*], 246–255, ACM, Manchester, England (July 1994).
- [12] Barker, K. J., Benner, A., Hoare, R., Hoisie, A., Jones, A. K., Kerbyson, D. K., Li, D., Melhem, R., Rajamony, R., Schenfeld, E., Shao, S., Stunkel, C., and Walker, P., "On the feasibility of optical circuit switching for high performance computing systems," in [*SC '05: Proceedings of the 2005 ACM/IEEE conference on Supercomputing*], 16, IEEE Computer Society, Washington, DC (Nov. 2005).
- [13] Artundo, I., Desmet, L., Heirman, W., Debaes, C., Dambre, J., Van Campenhout, J., and Thienpont, H., "Selective optical broadcast component for reconfigurable multiprocessor interconnects," *IEEE Journal of Selected Topics in Quantum Electronics: Special Issue on Optical Communication* **12**, 828–837 (July 2006).
- [14] Petracca, M., Lee, B. G., Bergman, K., and Carloni, L. P., "Design exploration of optical interconnection networks for chip multiprocessors," in [*Proceedings of the 16th IEEE Symposium on High Performance Interconnects*], 31–40 (Aug. 2008).

- [15] Sherwood-Droz, N., Wang, H., Chen, L., Lee, B. G., Biberman, A., Bergman, K., and Lipson, M., "Optical 4×4 hitless silicon router for optical networks-on-chip (NoC)," *Optics Express* **16**(20), 15915–15922 (2008).
- [16] Gu, H., Xu, J., and Zhang, W., "A low-power fat tree-based optical network-on-chip for multiprocessor system-on-chip," in [*Design, Automation and Test in Europe (DATE)*], 3–8 (Apr. 2009).
- [17] Greenfield, D. and Moore, S., "Fractal communication in software data dependency graphs," in [*Proceedings of the 20th ACM Symposium on Parallelism in Algorithms and Architectures (SPAA'08)*], 116–118 (June 2008).
- [18] Artundo, I., Heirman, W., Debaes, C., Loperena, M., Van Campenhout, J., and Thienpont, H., "Low-power reconfigurable network architecture for on-chip photonic interconnects," in [*17th IEEE Symposium on High Performance Interconnects*], 163–169 (Aug. 2009).
- [19] Vlasov, Y., Green, W. M. J., and Xia, F., "High-throughput silicon nanophotonic wavelength-insensitive switch for on-chip optical networks," *Nature Photonics* **2**, 242–246 (2008).
- [20] Assefa, S., Xia, F., and Vlasov, Y. A., "Reinventing germanium avalanche photodetector for nanophotonic on-chip optical interconnects," *Nature* **464**, 80–84 (Mar. 2010).
- [21] Heirman, W., Artundo, I., Carvajal, D., Desmet, L., Dambre, J., Debaes, C., Thienpont, H., and Van Campenhout, J., "Wavelength tuneable reconfigurable optical interconnection network for shared-memory machines," in [*Proceedings of the 31st European Conference on Optical Communication (ECOC 2005)*], **3**, 527–528, The Institution of Electrical Engineers, Glasgow, Scotland (Sept. 2005).
- [22] Heirman, W., *Reconfigurable Optical Interconnection Networks for Shared-Memory Multiprocessor Architectures*, PhD thesis, Ghent University (July 2008).
- [23] Magnusson, P. S., Christensson, M., Eskilson, J., Forsgren, D., Hallberg, G., Hogberg, J., Larsson, F., Moestedt, A., and Werner, B., "Simics: A full system simulation platform," *IEEE Computer* **35**, 50–58 (Feb. 2002).
- [24] Woo, S. C., Ohara, M., Torrie, E., Singh, J. P., and Gupta, A., "The SPLASH-2 programs: Characterization and methodological considerations," in [*Proceedings of the 22th International Symposium on Computer Architecture*], 24–36 (June 1995).
- [25] Heirman, W., Dambre, J., and Van Campenhout, J., "Synthetic traffic generation as a tool for dynamic interconnect evaluation," in [*Proceedings of the 2007 International Workshop on System Level Interconnect Prediction (SLIP'07)*], 65–72, ACM Press, Austin, Texas (Mar. 2007).
- [26] Green, W. M. J., Rooks, M. J., Sekaric, L., and Vlasov, Y. A., "Ultra-compact, low RF power, 10 Gb/s silicon mach-zehnder modulator," *Optics Express* **15**(25), 17106–17113 (2007).
- [27] Parhami, B., [*Introduction to Parallel Processing: Algorithms and Architectures*], Kluwer Academic Publishers (1999).
- [28] Pande, P. P., Grecu, C., Jones, M., Ivanov, A., and Saleh, R., "Performance evaluation and design trade-offs for network-on-chip interconnect architectures," *IEEE Transactions on Computers* **54**, 1025–1040 (Aug. 2005).
- [29] Lee, J., Kim, D., Ahn, H., Park, S., Pyo, J., and Kim, G., "Temperature-insensitive silicon nano-wire ring resonator," in [*Optical Fiber Communication Conference and Exposition and The National Fiber Optic Engineers Conference, OSA Technical Digest Series (CD)*], OWG4 (Mar. 2007).