PHOTONIC NOCS: SYSTEM-LEVEL DESIGN EXPLORATION

NETWORK-ON-CHIP IS A KEY ENABLING TECHNOLOGY TO ADDRESS THE CHALLENGES OF INTERCONNECTING THE INCREASING NUMBER OF CORES IN EMERGING CHIP MULTIPROCESSORS. BY LEVERAGING RECENT ADVANCES IN THE CMOS INTEGRATION OF PHOTONIC DEVICES AND THE UNIQUE PROPERTIES OF THE OPTICAL MEDIUM, PHOTONIC NOCS OFFER A PROMISING SOLUTION TO MEET THE COMMUNICATION REQUIREMENTS OF CHIP MULTIPROCESSORS WITH MINIMAL DRAW FROM THEIR POWER BUDGET.

Michele Petracca
Benjamin G. Lee
Keren Bergman
Luca P. Carloni
Columbia University

The new trend of integrating an increasing number of processing cores into a single die raises the importance of designing an efficient on-chip communication infrastructure. Packet-switched networks-on-chip (NoCs) are among the most promising paradigms for providing interconnect solutions in both general-purpose chip multiprocessors (CMP) and application-specific systems-on-chip (SoCs).

The design of NoC architectures introduces challenges in terms of offered throughput, layout, topology, and power efficiency.[1] Many studies have explored the optimization of NoC bandwidths and latencies, which directly impact system application performance. However, because packaging constraints will continue to impose strong limitations on the maximum on-chip temperature and power budget for the foreseeable future, the analysis and optimization of NoC energy efficiency becomes increasingly important as the number of cores on the chip grows. If fact, current CMP prototypes with tens of cores show that the power dissipated by the NoC accounts for more than 25 percent of the overall power.[2] Moreover, the power of NoCs implemented with current circuit techniques is estimated to be too high (by a factor of 10) to meet the expected needs of future CMPs.[1] Consequently, the limited on-chip power budget will have to be carefully distributed between computation and communication activities. Clearly, reducing the power dissipated by the NoC allows more of the limited power budget to be devoted to the cores, which directly improves the overall system's performance-per-watt.

In this context, photonic communication holds promise of providing a mechanism to realize large data transfers with minimal power dissipation. Several proposed architectures exploit silicon photonics for on-chip and off-chip communication (see the related sidebar). An NoC with photonic communication links offers two main advantages:

- the achievable communication bandwidth on a single waveguide (or link) can approach multiple terabits-per-second (Tbps) with limited power dissipation, and
- the power dissipation to first order is independent of the distance covered by the optical signal across the system

Published by the IEEE Computer Society

Silicon photonics for on-chip and off-chip communication

Recent advances in the CMOS integration of photonic devices have attracted the attention of several research groups to the application of such technologies to on- and off-chip communication. Kirman et al. study the performance improvement obtained using a CMOS-compatible photonic on-chip bus for future chip multiprocessors (CMPs).[1] Shacham et al. propose a hybrid approach to photonic on-chip interconnection.[2] They use an optical plane for high-bandwidth multiwavelength transmission links and a parallel electronic plane for the network management and control functions. We follow this architectural approach in this article. Elsewhere, Batten et al. discuss power-constrained processor-memory network architectures for future manycore systems.[3] Corona, Vantrease et al.'s 3D manycore architecture, uses photonic communication for both intercore communication and off-stack communication to memory or I/O devices.[4] Beausoleil et al. propose a high-performance manycore computing system divided into multiple silicon compute clusters.[5] Krishnamoorthy et al. review future opportunities for adopting photonic communication into a high-performance computing system at the chassis, chip package, and silicon microsystem levels.[6]

References

1. N. Kirman et al., ''On-chip Optical Technology in Future Bus-Based Multicore Designs,'' *IEEE Micro,* vol. 27, no. 1, Jan./Feb. 2007, pp. 56-66.
2. A. Shacham et al., ''On the Design of a Photonic Network-on-Chip,'' *Proc. Int'l Symp. Networks-on-Chip* (NOCS 07), IEEE Press, 2007, pp. 53-64.
3. C. Batten et al., ''Building Manycore Processor-to-DRAM Networks with Monolithic Silicon Photonics,'' *Proc. Hot Interconnects* (HOTI 08), IEEE CS Press, 2008, pp. 21-30.
4. D. Vantrease et al., ''Corona: System Implications of Emerging Nanophotonic Technology,'' *Proc. Int'l Symp. Computer Architecture* (ISCA 08), IEEE CS Press, 2008, pp. 153-164.
5. R.G. Beausoleil et al., ''A Nanophotonic Interconnect for High-Performance Many-Core Computation,'' *Proc. Hot Interconnects* (HOTI 08), IEEE CS Press, 2008, pp. 182-189.
6. A.V. Krishnamoorthy et al., ''Optical Interconnects for Present and Future High-Performance Computing Systems,'' *Proc. Hot Interconnects* (HOTI 08), IEEE CS Press, 2008, pp. 175-177.

and scales only with the link transmission interface circuitry (modulators, drivers, and receivers).

The effective lack of optical memories or equivalent optical RAM and the impracticality of processing directly in the optical domain will force designers to combine photonic communication with electronic control. However, although the integration of optical devices on a chip still presents many challenges, recent years have seen remarkable breakthroughs in the field of CMOS-compatible silicon photonics (see the ''Physical-layer components'' sidebar). Building on these advances, Shacham et al. proposed a photonic NoC for CMPs based on a hybrid approach: a high-bandwidth circuit-switched photonic network combined with a low-bandwidth packet-switched electronic network.[3] The electronic network carries small control (and data) packets, whereas the photonic network transfers large data messages between core pairs. The NoC operates as follows:

1. A source core reserves a photonic circuit by sending a path-setup packet over the electronic network to the destination core, which replies with a short Ack pulse over the photonic network (*path-setup process*).
2. The source sends the data over the photonic circuit, which can offer near Tbps of photonic transmission line-rate per core by combining time-division and wavelength-division multiplexing (TDM-WDM).
3. At the end of the communication, the source releases the photonic circuit by transmitting a tear-down packet (*path teardown process*).

Figure 1a shows the main organization of the photonic NoC for a 16-core CMP; Figure 1b shows Shacham et al.'s internally nonblocking photonic switch.[4] The photonic switches let every core access the network, but the considered network can't simultaneously sustain all possible communications among distinct cores because internal congestion can occur during the photonic paths' setup. In fact, the network's blocking topology offers limited connectivity. We refer to this topology as the *blocking torus*. Figure 1c shows a possible layout for a 16-core CMP. To optimize the fabrication process, the

Physical-layer components

Recent breakthroughs in silicon photonic integration have resulted in a large toolbox of CMOS-compatible photonic components required for constructing simple photonic NoCs (modulators, links, switches, receivers, and so on). Because of fabrication and design improvements, sub-μm-dimensional photonic links now typically achieve propagation losses around 1.7 dB/cm and off-chip coupling losses of 0.5 dB/facet. The microring (or microdisk) resonator leverages tremendous functionality, acting as a modulator, switch, or wavelength multiplexer. The resonator reaches modulation speeds in excess of 10 Gbps with 85 fJ/b of measured power dissipation, with much lower theoretical power requirements.[1,2] Furthermore, researchers have demonstrated simple microring switches with throughput bandwidths of 250 Gbps, easily scalable to more than 1 Tbps.[3,4] The entire four-port nonblocking switch in Figure 1b (main article) has also been fabricated and characterized, exhibiting multiwavelength routing functionality through thermally tuned and stabilized microheaters.[5] Finally, receivers using SiGe or Ge photodetectors with CMOS postamplifier circuitry have also achieved promising results, demonstrating near-pJ/b energy dissipation in addition to high data-rate operation at 15 Gbps.[6] Although more improvements are expected in CMOS-compatible receiver designs, many researchers also foresee receiverless operation of SiGe detectors as a feasible means to drastically reduce the energy/bit contribution of the opto-electronic conversion.

References

1. Q. Xu et al., ''12.5 Gbit/s Carrier-Injection-Based Cilicon Microring Silicon Modulators,'' *Optics Express,* vol. 15, no. 2, 2007, pp. 430-436.
2. M.R. Watts et al., ''Ultralow Power Silicon Microdisk Modulators and Switches,'' *Proc. IEEE Conf. Group IV Photonics,* IEEE Press, 2008, pp. 4-6.
3. A. Biberman et al., ''250 Gb/s Multi-Wavelength Operation of Microring Resonator-Based Broadband Comb Switch for Silicon Photonic Networks-on-Chip,'' *Proc. European Conf. Optical Comm.,* IEEE Press, 2008, pp. 1-2.
4. Y. Vlasov, W.M.J. Green, and F. Xia, ''High-Throughput Silicon Nanophotonic Wavelength-Insensitive Switch for On-Chip Optical Network,'' *Nature Photonics,* vol. 2, Mar., 2008, pp. 242-246.
5. B.G. Lee et al., ''Multi-Wavelength Message Routing in a Non-Blocking Four-Port Bidirectional Switch Fabric for Silicon Photonic Networks-on-Chip,'' *Optical Fiber Comm. Conf.,* Optical Soc. of America, 2009, paper OMJ4.
6. S.J. Koester et al., ''Ge-on-SOI detector/Si-CMOS Amplifier Receivers for High-Performance Optical-Communication Applications,'' *J. Lightwave Technology,* vol. 25, no. 1, 2007, pp. 46-57.

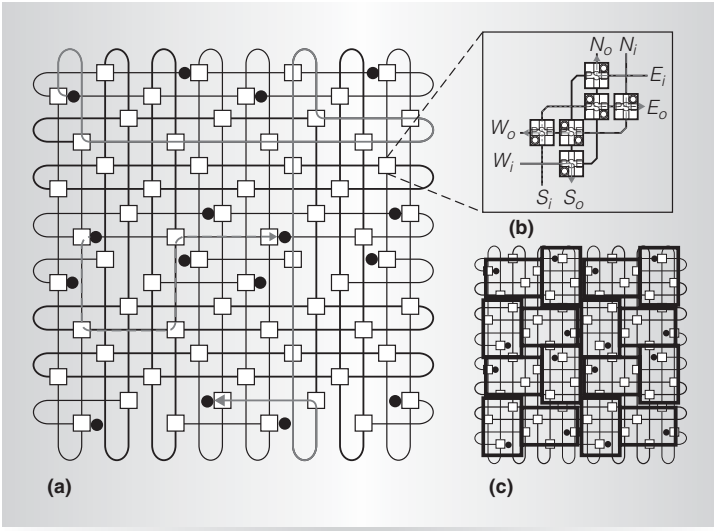

Figure 1. A 16-core blocking torus photonic NoC (a).[3] The dashed line represents the shortest path; the solid line represents the longest. A basic nonblocking switch (b).[4] A possible core layout over the NoC (c). In (a) and (c) the black circles represent the cores' network interfaces (gateways) and the white squares represent the photonic switches.

NoC's photonic devices and the CMP cores will likely sit on different planes using 3D integration (3DI).[5]

Nonblocking topologies

From a conceptual viewpoint, there are pros and cons to using nonblocking topologies for photonic NoCs. A strictly nonblocking network can simultaneously handle the maximum number of connections, thus reducing the blocking probability in setting up a connection due to network congestion. Indeed, a nonblocking topology guarantees that a block occurs only if two sources want to communicate with the same destination, and never due to a lack of internal available resources. In our proposed hybrid photonic NoC approach, a communication setup represents an overhead, because no data are transferred during that time. For every blocked communication, the path-setup procedure is interrupted and repeated successively until the destination is reachable and the data can be transferred. Therefore, the

lower the blocking probability, the lower the average number of path-setup attempts per communication. Fewer setup attempts means lower overhead for establishing a path between a source and destination, and thus higher throughput and lower latency.

Here, we compare alternative nonblocking topologies with the blocking torus in terms of scalability and performance. In particular, we propose two nonblocking topologies: a *crossbar* and a *nonblocking torus*. Both are strictly nonblocking with $O(N^2)$ complexity in terms of the number of switches, where N is the number of cores, and both use 4×4 switches, such as the switch in Figure 1b. However, the disposition of the routers and gateways impacts the performance benefits of adopting a nonblocking topology as opposed to a blocking one.

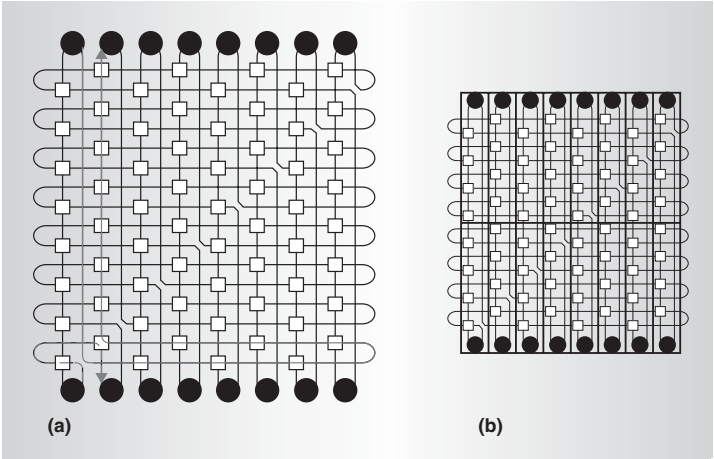

Figure 2. A 16-core crossbar (a), and a possible core layout over the NoC (b). Black circles represent core gateways and white boxes represent switches.

Crossbar

Figure 2a shows a crossbar topology for a 16-core CMP. The switches are organized in an 8×8 matrix and connected by bidirectional links. For clarity, we use 16-core CMPs for the topology pictures but conduct the performance analysis over 36-core CMPs. Each pair of facing gateways on a column share a row for injection and a column for ejection, thereby exploiting the bidirectionality of the 4×4 switches. Crossbars have limited scalability in terms of both resources needed to build the network and maximum (and average) *path length*—that is, the number of hops between two communicating cores. The maximum path length impacts the maximum attenuation experienced by the photonic signal. The average path length affects the average duration of the path-setup process, thus impacting the average throughput and latency performance.

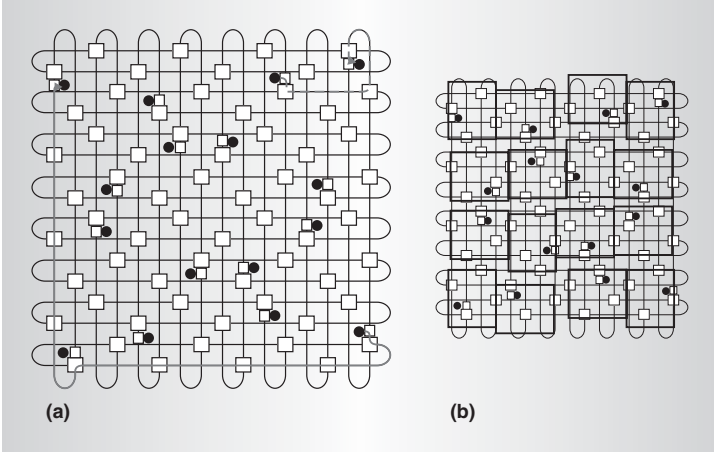

Figure 3. A 16-core nonblocking torus (a), and a possible core layout over the NoC (b).

Nonblocking torus

In a nonblocking torus, the number of switches is at least $N/4$ times the number of gateways. Because the links and switches are bidirectional, a nonblocking torus can have at the most two gateways injecting on each row and two gateways ejecting from each column. Figure 3 shows a nonblocking torus for a 16-core CMP.

To implement this topology, we divided the chip into four quadrants, so N must be a multiple of 4. Each quadrant is a square matrix of switches, where $N/4$ gateway switches are placed on the diagonal, so that one gateway switch is placed on each row and one gateway switch on each column. The two horizontally contiguous quadrants are identical. The difference between the quadrants of the upper and bottom halves is that the gateway switches are placed above or below the corresponding row, based on the injection rule. The resulting topology is a torus with $N^2/4$ switches and

Table 1. Comparison of the three topologies with an increasing number of cores.

	Blocking torus		Crossbar		Nonblocking torus	
Cores	Switches	Average path	Switches	Average path	Switches	Average path
16	144	8	64	12	80	6
36	324	12	324	27	360	11
64	576	16	1024	48	1088	18

N gateway switches. Figure 3 is the folded version of this network, in which the distance remains constant between the switches and every column and row.

During ejection, a message passing through a column can enter a gateway switch from either vertical port. If this message is destined for the attached core, it's deflected toward the gateway. During injection, the gateway sends a packet to the gateway switch, which forwards it to the nearest row. Once the packet is on the row, it simply follows an XY minimum-distance routing algorithm: it reaches the right column passing through the input row and then passes through the output column to reach the destination gateway switch.

Although the complexity in terms of the asymptotic number of switches grows as $O(N^2)$, the nonblocking torus offers a remarkable improvement with respect to the crossbar by reducing the average path length between cores. Table 1 reports the number of switches and the average electronic path lengths, expressed in number of switch-to-switch hops, for the three alternative NoC topologies. The lack of scalability also impacts the nonblocking networks' performance gain with respect to the blocking ones. As the number of switches increases, so does the time needed to set up the path. Therefore, from the viewpoint of delivered performance per number of deployed photonic devices, nonblocking networks are best suited for connecting a limited number of cores. The topology scalability issues mainly affect the area requirements for placing the devices composing the photonic interconnect; however, physical-layer scalability problems related to signal integrity and actual system feasibility also arise (see the "Physical-layer scalability" sidebar).

Topology comparison

We performed a comparative analysis of the blocking torus, crossbar, and nonblocking torus topologies using simulations under different traffic patterns with the Photonic On-chip Interconnection Network Traffic Simulator (POINTS).[3]

Uniform traffic

We evaluated throughput-per-core as the ratio of the time during which a core is transmitting photonic messages on the NoC to the total simulation time. This metric is a function of the average path-setup overhead, which depends on the NoC topology and the ratio of the average photonic message size to the photonic transmission line rate. The *offered load* is the ratio of the time when a core is ready to transmit at least one message to the total simulation time. In a noncongested network, the throughput-per-core matches the offered load. We assume 36 cores exchanging DMA transfers of fixed size, equal to 16 kbytes, with a line-rate of 960 gigabits per second. This corresponds to a photonic message with a duration of 134 ns. We assume the TDM-WDM message to be composed of 24 wavelength channels, each operating at 40 Gbps. For lower data rates (for example, 10 Gbps), we use spatial-division multiplexing (four parallel waveguides).

By introducing photonic NoCs, we aim to provide high-bandwidth, low-latency communication channels for large data transfers between cores. They're therefore suitable for applying traffic-aggregation policies. In this case, each core acts as a multithreaded processor that executes many threads in parallel, and each thread can independently request a data transfer to a thread running on another core. Therefore, in this analysis, a core is a

Physical-layer scalability

Many issues in addition to topology warrant design exploration. Photonic NoC designers face many challenges as they search for an optimal arrangement of network devices and components with which to optimize performance at both the physical and network layers. NoC designers can increase theoretical throughput and latency performance by implementing sophisticated topologies. However, the added complexity can stress the physical layer performance if not taken into account. Considerations should include the effects of the aggregated insertion losses and power penalties associated with the signal passing through the switches, waveguide crossings, and link distances, as well as the effects of the inadvertent optical power that can leak through the wrong port of a switch or reflect backward from a waveguide crossing.

As Table 1 in the main article notes, the number of switches that a signal encounters scales dissimilarly among topologies. Given the signal power losses and the addition of crosstalk within the network, scaling a particular topology beyond a certain number of nodes might result in the inability to receive the optical message reliably. In some cases, the performance of one of the simplest photonic components—a waveguide crossing—is the most crucial because of the large number that can be encountered in an optical pathway. Other considerations might account for the effects of thermally varying environments on the photonic elements' performance, or the effect of scaling the number of wavelengths per link given a particular nonlinear power threshold within the optical waveguide. Ultimately, the physical layer can't be completely separated from the networking layer. It's therefore necessary to evaluate the photonic NoC performance using a physical-layer simulator working in tandem with a network simulator.[1] Such a design environment would provide the necessary tools for developing a fully maximized photonic NoC to meet the challenging requirements of the chip multiprocessor industry.

Reference

1. J. Chan et al., ''Insertion Loss Analysis in a Photonic Interconnection Network for On-Chip and Off-Chip Communications,'' *Proc. Ann. Meeting Lasers Electro-Optics Soc.,* IEEE Press, 2008, pp. 300-301.

traffic source simultaneously scheduling multiple communication requests to different destinations. The network can serve these communications one at a time because they all share the same gateway. We model the maximum number of simultaneous communications requests managed by a core to be the number of threads running on that core.

Figure 4a shows the throughput-per-core as a function of the offered load for four distinct scenarios:

- blocking torus with single-thread cores;
- blocking torus with multithread cores;
- crossbar with multithread cores; and
- nonblocking torus with multithread cores.

Using multithread cores lets us better exploit the high bandwidth offered by a photonic NoC, leading to a gain of more than 26 percent in throughput-per-core. In fact, a single-thread core can make one communication request at a time, stalling the thread until the request isn't served. However, whenever the path setup for that request is blocked in the NoC, the communication can't take place until the network congestion is over, and thus loses efficiency. Having more parallel threads in the same core allows more requests. Then, when one request can't be served, another might be more successful because it's addressed to destinations in less congested parts of the network.

Surprisingly, analysis of the relationships among topologies using the same core model shows similar performance for crossbar and blocking torus. Generally, a nonblocking topology doesn't achieve a near-100-percent maximum throughput per core because the overhead introduced by the path-setup process isn't negligible for short-duration messages. On the other hand, by definition, a nonblocking topology guarantees the delivery of a message to every free destination. This advantage, however, is partially neutralized in the crossbar because the long electronic paths considerably increase the propagation time of the path-setup packet over the control network. In the nonblocking torus, the distance between two gateways is comparable to the corresponding distance in the blocking torus, leading to a throughput gain of about 13 percent.

The latency is the time elapsed from the generation of the thread's transfer request to the reception of the last bit of the photonic message at the destination core. The latency experienced by the photonic messages mainly depends on the path-setup process. The optical delay is negligible, so the time a request must wait to be completely satisfied

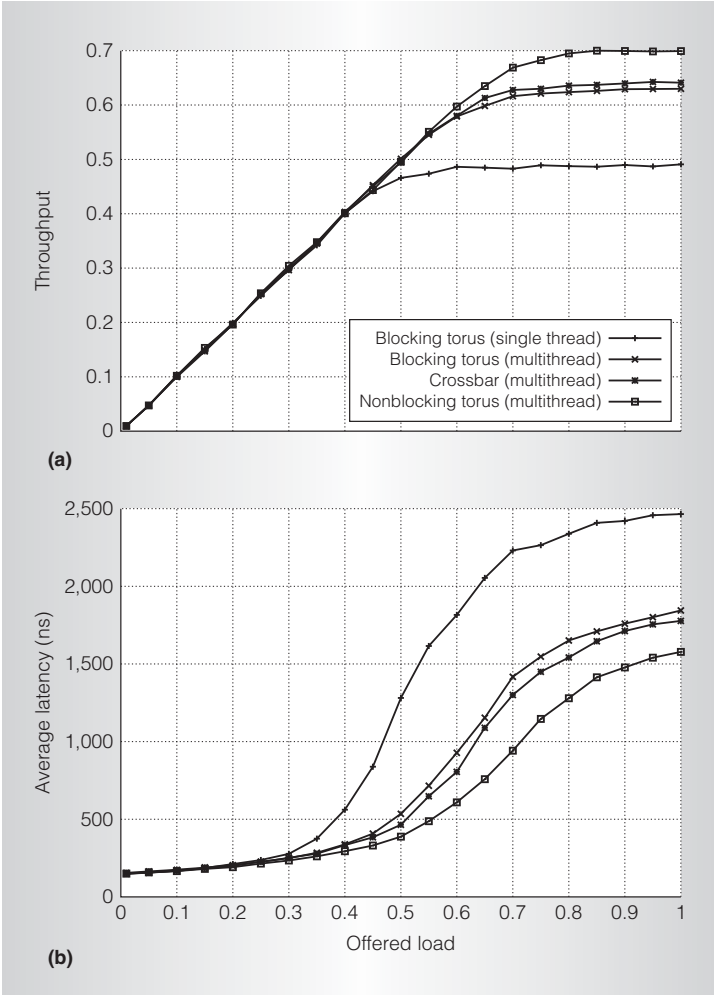

Figure 4. Comparison of various 36-core NoC topologies: throughput-per-core (a), and average latency per node (b).

is basically the duration of the path setup plus that of the photonic message. In our simulations, the message duration is constant and equal for all topologies. The latency is strictly related to the global throughput, because the longer the average time to reserve a path, the lower the average number of active connections over the network.

Figure 4b compares the average latency over all the communications that took place on the network for the offered load. The higher the traffic load, the more likely a connection can't be established because the destination is already receiving other data, or because of in-network congestion in the case of blocking networks. Note that the latency curves don't diverge but saturate because the core model has backpressure at the traffic source. Under overload conditions, the source stops sending packets. In this way, we model the possibility of stalling the threads until their communication request has been served, thus modeling a finite number of computing threads.

Nonuniform traffic

We also performed a comparative analysis under nonuniform traffic shapes. We kept the same hypotheses for message duration, line rate, and number of cores, but we estimated the throughput per core under Tornado, Transpose, and 6 Hot Spot traffic shapes. For brevity, Table 2 lists only the network maximum throughput offered by each topology for all simulated traffic shapes. As discussed earlier, average communication latency and throughput are strictly related.

The results for uniform traffic shown in Table 2 match the saturation points of the curves in Figure 4. The table also shows that a nonblocking torus topology delivers even higher gain (up to 53 percent) than a blocking torus or crossbar for Tornado and Transpose traffic. In these cases, each destination receives messages from the same source, which lets the cores fully exploit the interconnect's nonblocking property. Again, for the crossbar the long average path reduces the offered performance.

The 6 HotSpot traffic consists of 30 out of 36 cores sending and receiving traffic uniformly to and from the other six cores. This emulates a shared memory scenario in which 30 processors access six memory banks. In this case, 83 percent of the offered load is addressed to six destination nodes. Because the traffic shape is asymmetric, a nonblocking topology provides negligible advantages because the congestion is mainly due to communications sharing the same destination and not to in-network blocking. The crossbar's performance is even worse than the blocking torus's.

Case study: FFT computation

Fast Fourier transform (FFT) can exploit the large bandwidth offered by photonic on-chip communication. Using POINTS,

Table 2. Maximum offered throughput (%) for four kinds of traffic and three topologies (36 multithread cores).

Topology	Uniform	Tornado	Transpose	6 HotSpot
Blocking torus	62	58	70	28.5
Crossbar	63	85	83	28.0
Nonblocking torus	70	89	86	29.5

we analyzed the execution of the classic Cooley-Tukey FFT algorithm running on a hypothetical CMP built in a future 22-nm technology process. Assuming classic scaling and a chip size of about 625 mm^2 we should be able to integrate 36 cores as complex as the first generation of the IBM Cell multicore processor into our CMP. Assuming the use of 3DI[5] to combine a processing core plane with an optical NoC plane and various on-chip memory planes, each core could have access to a local memory of about 0.5 Gbytes.

The FFT computation runs on 32 of the 36 available cores. In the first phase, each core processes $k = m/M$ sample elements, where m is the size of the array of input samples and M is the number of cores. Next, the algorithm proceeds with a sequence of $\log M$ iterations. At each iteration a computation step follows a communication step, during which the processors exchange data according to a butterfly scheme. Specifically, at each iteration, a core

- sends a copy of the subarray resulting from the previous computation to another core X, keeping a local copy;
- simultaneously receives another subarray from core X; and,
- when both transfers are complete, linearly combines the local copy and newly received subarray.

At the end of all iterations, the result of the FFT on the original m-elements input vector is the merge of the k-element portions of subarrays resulting from the local computation in each core. The time to perform the FFT is the sum of the time for the computation, which depends on the core architecture, plus the time to move the data among the cores. The last component depends on the line rate and the topology. The line rate influences message duration, whereas the topology influences the average number of attempts to deliver a data subarray.

The current Cell processor reportedly computes a large single-precision FFT (2^{24} samples) in 43 ms using Bailey's FFT algorithm.[6] We assume that each core in our CMP corresponds to a future version of the Cell whose internal processing units (today's Synergistic Processing Elements) have twice the amount of local-store memory and a double precision floating-point unit. This would let us scale the same result Chow et al.[6] describe to a 256-Mbyte array of 2^{24} double-precision sample elements and thus use Bailey's FFT algorithm within each core to complete the first phase of the Cooley-Tukey algorithm in about 43 ms. Starting from this number, and given that the Bailey's algorithm requires $5k \log k$ floating-point operations, the computation step should take 1.8 ms in each subsequent iteration, for $k = 2^{24}$. Finally, assuming a 960-Gbps photonic transmission line rate, our CMP equipped with the nonblocking torus would execute a 2^{29} double-precision sample FFT in about 66 ms, of which 14 ms are needed for the butterfly data exchanges. We'll consider 66 ms the reference value for the total execution time in the rest of our analysis.

For an application with such a regular communication pattern, a nonblocking topology lets all transfers take place simultaneously because in each butterfly stage each core communicates with a different destination. A blocking topology, on the other hand, presents some conflicts within the network, thus forcing some communications to wait for others to complete. In our simulations, the same CMP equipped with a blocking torus takes 74.6 ms to complete the FFT computation due to an increase of 8.6 ms for the butterfly data exchanges with respect to the nonblocking topology.

For the remainder of this article, we use the nonblocking torus with a 625-mm^2 square die as a reference topology. Hence, a hop between two switches spans about 2.78 mm, and the average path between two gateways is 11 hops with four turns.

Because silicon photonics represent a new technology, and thus there are no well-established roadmaps, it's difficult to predict the future scaling of device power consumption. We therefore consider a conservative scenario for the photonic link power requirements. The performance of currently available photonic transceivers in 130-nm CMOS technology[7] suggests that the energy consumption of a complete photonic connection in our 22-nm chip could be 0.8 pJ/b or lower. Hence, to simultaneously transfer 32 256-Mbyte blocks at 960 Gbps, we need less than 24.5 W (drawing on the conservative prediction)—that is, about 770 mW per connection. The total power value remains the same for the blocking torus, in which there are 12 hops and three turns, because most of the power dissipation is in the optical interfaces rather than the polarized rings.

After evaluating the CMP's performance with a photonic network, we tried replacing it with an equivalent electronic network. Because channel utilization for the FFT subarray transfers persists across cores, a circuit-switched data network achieves better performance than a packet-switched NoC. Hence, we assume the same organization as in the photonic NoC. Further, we conservatively assume that the electronic equivalent of a photonic switch (Figure 1b) is ideal—that is, without any delay and power consumption. To evaluate communication delay and power consumption over the equivalent electronic circuit-switched NoC, we consider optimally repeated wires in the given 22-nm technology design point and assume a value of 0.25 pJ/b/mm for their energy consumption.[1] We note that this is a simple single-point comparison performed to extract the critical metrics differentiating the photonic and electronic designs and of course doesn't represent the myriad possible electronic circuitry designs.

Given the amount of data to be moved during the transfer stages, the message duration is at least of the order of milliseconds. Because the path-setup time and light propagation are tens and fractions of nanoseconds, respectively, we consider them negligible. These considerations are valid for an electronic data plane as well: even if the signal propagation is slower in the copper than in an optical waveguide, the latency is still around a few nanoseconds. In summary, the computation time doesn't depend on the media, but only on the line rate. For a given topology, computation time is the same for both the photonic network and an electronic equivalent network as long as the line rate is the same.

To assess the gain in performance-per-watt offered by the photonic NoC, we consider two cases:

- To achieve the same execution time as the photonic NoC, the electronic NoC must operate at the same line rate, but in doing so it dissipates 7.6 W per connection, a value about 10 times higher. This leads to an overall power dissipation for the electronic NoC of about 244 W, a value that alone would exceed the total power budget for the CMP.
- To achieve the same power dissipation, the electronic NoC must operate at a line rate of 100 Gbps, a reduction of 90 percent, thereby taking about 190 ms to complete the FFT computation. This is three times more than the reference network because the computation time remains the same.

Note that in our scenario a blocking topology has at best the same power efficiency as a nonblocking one. The data to be transferred during the algorithm's execution depend on the application and the energy required to transfer a bit of information from source to destination. This is equivalent for both topologies, because it depends on the chip size in the electronic domain and on the line rate in the photonic domain. Therefore, both topologies need the same amount of energy for data transfers. However, the nonblocking topology can deliver data more quickly because it's congestion free for the given traffic pattern. The blocking

topology has a lower average power dissipation but also lower performance, making the efficiency the same.

Both topologies offer the same peak power consumption. Indeed, in the last stage of the algorithm, when every supercore communicates with its neighbor, the blocking topology can host all the simultaneous communications as well. Because the design and dimension of the chip's cooling system is generally based on the peak power consumption, the two topologies can be considered equivalent. However, a nonblocking topology will deliver better performance in terms of algorithm execution time.

Our analysis shows how, under the given projections, photonics can deliver advantages in terms of power dissipation. If we consider a broader design space for our case study, Figure 5 shows the scaling of the power gain with different projections of the future power dissipation of photonic transceivers and electronic optimally repeated wires. We examine the energy efficiencies that the photonic transceivers would require to realize performance gains over equivalent electronic NoCs under a range of possible electronic designs. For example, under the original assumption of 0.25 pJ/b/mm for the optimally repeated electronic wires, the photonic NoC could realize 40× performance-per-watt gains if the optical transceivers could deliver 0.2 pJ/b for the associated communications energy consumption. On the other hand, more aggressive electronic circuitry designs realizing energy efficiencies of 0.08 pJ/b/mm would reduce the possible advantages of photonic NoCs for on-chip communications to factors less than 10×.

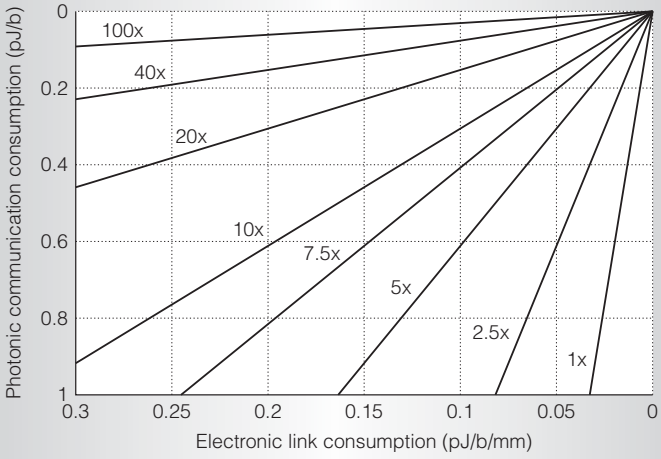

Figure 5. FFT case study. Power savings in system-level communication for different values of the link dissipation (at 22 nm).

Although semicustom NoCs that dissipate low power (that is, hundreds of mW) can be efficiently built for SoCs used in embedded applications, NoCs based on traditional circuit techniques might not satisfy the bandwidth requirements and die area of future high-performance CMPs. Although some promising new electronic circuit techniques could possibly achieve these power and bandwidth requirements, our experimental results show that photonic communication can potentially deliver the necessary reductions in power consumption in a scalable fashion, particularly for critical applications that require massive data transfers at high bandwidths over a large-die CMP. Admittedly, the complexity of photonic integration remains high as the technology is significantly less mature than electronics. On the other hand, photonic communication across multiple chips and to DRAM memories can provide the same bandwidth-per-watt as on-chip communication independently of the distances spanned connecting elements within an entire multiblade system.[8] The introduction of photonic I/O circuitry can then pave the way to the introduction of hybrid NoCs for CMPs to simultaneously reduce power consumption on and off chip. MICRO

References

1. J.D. Owens et al., ''Research Challenges for On-chip Interconnection Networks,'' *IEEE Micro,* vol. 27, no. 5, Sept./Oct. 2007, pp. 96-108.
2. Y. Hoskote et al., ''A 5-GHz Mesh Interconnect for a Teraflops Processor,'' *IEEE Micro,* vol. 27, no. 5, Sept./Oct. 2007, pp. 51-61.
3. A. Shacham, K. Bergman, and L.P. Carloni, ''Photonic Networks-on-Chip for Future Generations of Chip Multiprocessors,''

IEEE Trans. Computers, vol. 57, no. 9, 2008, pp. 1246-1260.
4. A. Shacham et al., ''Photonic NoC for DMA Communications in Chip Multiprocessors,'' *Proc. 15th IEEE Symp. High-Performance Interconnects* (HOTI 07), IEEE CS Press, pp. 29-38.
5. W. Haensch, ''Why Should We Do 3D Integration?'' *Proc. 45th Design Automation Conference (DAC 08),* IEEE Press, 2008, pp. 674-675.
6. A.C. Chow, G.C. Fossum, and D.A. Brokenshine, ''A Programming Example: Large FFT on the Cell Broadband Engine,'' IBM white paper, 2005; http://www.t-platforms.ru/pdf/GSPx_FFT_paper_legal_0115.pdf.
7. C. Schow et al., ''A < 5mW/Gb/s/link, 16 × 10 Gb/s Bi-directional Single Chip CMOS Optical Transceiver for Board-level Optical Interconnects,'' *Proc. IEEE Int'l Solid-State Circuits Conf.* (ISSCC 08), IEEE Press, 2008, pp. 294-295.
8. B.E. Lemoff et al., ''MAUI: Enabling Fiber-to-the-Processor with Parallel Multiwavelength Optical Interconnects,'' *J. Lightweight Technologies,* vol. 22, no. 9, Sept. 2004, pp. 2043-2054.

Michele Petracca is a postdoctoral researcher in the Computer Science Department at Columbia University. His research interests include optical communication and networks, system and RTL design for network devices, and networks-on-chip design. Petracca has a PhD in electrical engineering from Politecnico di Torino, Italy. He is member of the IEEE.

Benjamin G. Lee is a postdoctoral researcher at the IBM T.J. Watson Research Center, Yorktown Heights, New York. His research interests include silicon photonic devices, integrated optical switches and networks for high-performance computing systems, and all-optical processing systems. Lee has a PhD in electrical engineering from Columbia University. He is a member of the IEEE Photonics Society and the Optical Society of America.

Keren Bergman is a professor in the Department of Electrical Engineering at Columbia University, where she directs the Lightwave Research Laboratory. Her research interests include optical interconnection networks for advanced computing systems, photonic packet switching, and nanophotonic networks-on-chip. Bergman has a PhD in electrical engineering from the Massachusetts Institute of Technology. She is the coeditor in chief of the newly launched *IEEE/OSA Journal of Optical Communications and Networking* and is a Fellow of the IEEE and of the Optical Society of America.

Luca P. Carloni is an associate professor in the Department of Computer Science at Columbia University. His research interests include design tools and methodologies for integrated circuits and systems, distributed embedded systems design, and the design of high-performance computers. He has a PhD in electrical engineering and computer sciences from the University of California, Berkeley. He is a member of the IEEE and the ACM.

Readers can contact Michele Petracca at the Dept. of Computer Science, Columbia Univ., 1214 Amsterdam Ave., New York, NY 10027; petracca@cs.columbia.edu.