CONCURRENCY AND COMPUTATION: PRACTICE AND EXPERIENCE
Concurrency Computat.: Pract. Exper. (2014)
Published online in Wiley Online Library (wileyonlinelibrary.com). DOI: 10.1002/cpe.3332

SPECIAL ISSUE PAPER

Managing resources dynamically in hybrid photonic-electronic networks-on-chip

Antonio García-Guirado[1,*,†], Ricardo Fernández-Pascual[1], José M. García[1] and Sandro Bartolini[2]

[1]*Departamento de Ingeniería y Tecnología de Computadores, University of Murcia, 30100 Murcia, Spain*
[2]*Dipartimento di Ingegneria dell'Informazione, University of Siena, 53100 Siena, Italy*

SUMMARY

Nanophotonics promises to solve the scalability problems of current electrical interconnects thanks to its low sensitivity to distance in terms of latency and energy consumption. Before this technology reaches maturity, hybrid photonic-electronic networks will be a viable alternative. Ideally, ordinary electrical meshes and ring-based photonic networks should cooperate to minimize overall latency and energy consumption, but currently, we lack mechanisms to do this efficiently. In this paper, we present novel fine-grain policies to manage the photonic resources in a tiled chip multiprocessor (CMP) scenario. Our policies are dynamic and base their decisions on parameters such as message size, ring availability, and distance between endpoints, at the message level. The resulting network behavior is also fairer to all cores, reducing processor idle time thanks to faster thread synchronization. All these policies improve performance when compared to the same CMP without the photonic ring, and the most elaborate ones reduce the overall network latency by 50%, execution time by 36%, and network energy consumption by 52% on average, in a 16-core CMP for the PARSEC benchmark suite. Larger hybrid networks with 64 endpoints for 256-core CMPs, based on Corona and Firefly designs, also show far superior throughput and lower latency if managed by one of the proposed policies. Copyright © 2014 John Wiley & Sons, Ltd.

Received 27 May 2014; Revised 28 June 2014; Accepted 11 June 2014

KEY WORDS: on-chip photonics; cache coherence

1. INTRODUCTION

To be able to keep pace with Moore's Law, the latest generations of most microprocessors have adopted an on-chip multi-core architecture where the last-level cache (LLC) is typically distributed across tiles [1]. This configuration enables scalability, but while each core can directly access the portion of cache in its own tile, it needs to use an interconnection network to access the cache resources in other tiles. The tiled-CMP design paradigm is devised to run complex and heterogeneous multi-threaded applications, which require efficient communication and synchronization between threads within the chip. This leads to the need for efficient on-chip interconnection mechanisms such as high-performance and structured networks-on-chip (NoCs) for interconnecting the tiles (each one including typically cores and cache resources).

The execution time of applications is becoming more and more affected by network traffic and particularly by the average distance traversed to retrieve data from the correct LLC tile in the chip (number of network hops). As the core count increases, the number of retransmissions that messages

*Correspondence to: Antonio García-Guirado, Departamento de Ingeniería y Tecnología de Computadores, University of Murcia, 30100 Murcia, Spain.
†E-mail: toni@ditec.um.es

suffer in the electrical network also increases, compromising the scalability of future chip multiprocessors. In addition, data transmission through the on-chip interconnect is starting to account for most of the energy consumption of a chip; a scenario that is expected to get worse [2, 3]. This problem must be addressed to continue increasing the performance of future chips within a reasonable power budget.

Advances in silicon photonics have enabled the integration of optical interconnects inside silicon chips [4, 5]. This disruptive technology provides low-energy fast data transmission across the whole chip regardless of distance, and can be a solution to the scalability problems of NoCs. For instance, transmitting information between two opposite corners of an 8×8 electronic mesh at 4 GHz requires traversing 15 routers and 14 inter-tile links, taking up tens of processor cycles (also at 4 GHz). Traversing the same distance in a photonic waveguide (group velocity of light into silicon is about 15 ps/mm) can take as little as two processor cycles, needs no retransmissions, and uses significantly less energy in the process.

Since silicon-photonic integration became feasible for CMP interconnects, a myriad of photonic networks have been proposed. Many topologies have been studied, from simple photonic rings [6–8] that operate like a crossbar, to complex articulated topologies [9, 10] that require or combine different transmission technologies, and to logical all-to-all interconnect designs [11]. Some complex photonic interconnects use supporting circuit-establishing electrical networks [9, 10], which limit severely the latency and energy advantages of nanophotonics in scenarios like hardware-cache-coherent CMPs. We focus on a simple photonic structure (ring) instead and show that, if properly managed, it can potentially deliver large latency and energy improvements without needing big investments in complex topologies and/or organizations.

Investigation on the maximum benefits achievable from simple optical topologies (e.g., ring-based) is strategic because, for relatively short-term solutions, the use of a simple 3D-stacked photonic network can be an interesting design choice, especially for pursuing low power.

Hybrid photonic-electronic NoCs attempt to make the most of both transmission technologies [12–15]. While in the classic electronic heterogeneous network scenario, the low-latency network was very power hungry and was used selectively for accelerating certain messages; in this new scenario, the low-latency raw photonic technology enables the faster and less power consuming network of the system in terms of dynamic power. But, when the load increases, it can suffer from long message queuing because of serialization and contention, reducing its potential benefits. Furthermore, increasing photonic resources to limit these serialization effects would introduce more waveguide crossings, thus higher insertion loss and laser power, which would in turn further increase the static power consumption. Moreover, for very short distances (e.g., 1 hop) also, the latency advantage of photonics over electronics can be quite limited because of conversion overhead.

For these reasons, hybrid photonic-electronic NoCs, especially those based on simple physical topologies, which will soon be implementable, need to be carefully managed to take advantage of the latency and energy advantages of photonic technology. Currently, there is a lack of adequate policies to carry out this management. Our purpose is to develop mechanisms to make the best use of a photonic network that works in cooperation with an electrical mesh in a modern CMP. To our knowledge, these are the first proposed ad-hoc management strategies that use real-time information for hybrid photonic-electronic NoCs at the message level. We test these policies on CMPs equipped with different hybrid network designs. First, we consider a simple single-waveguide photonic ring, which is a likely representative of near-future CMPs, showing that only an appropriate policy is able to reduce the average execution time of the PARSEC benchmarks by 36% with respect to an electrical mesh while network energy consumption is also reduced by 52% on average. However, the proposed mechanisms are applicable, with appropriate tuning, to different network topologies, sizes, and arbitration mechanisms. To prove so, we also test the policies on larger and more complex networks, showing large throughput and latency benefits.

Furthermore, our results show that our more advanced policies significantly outperform naïve ones both in execution time and power consumption, demonstrating the importance of dynamic electro/photonic message management policies. Methodologically, considering first a simple one-waveguide network allows dissecting and tuning policies in insulation from other effects that might occur in more complex network organizations.

DOI: 10.1002/cpe

The main contributions of this work can be summarized as follows:

- We propose novel policies to efficiently use photonic resources and test them by enabling a profitable usage of a simple photonic ring in collaboration with an electrical network.
- We evaluate these policies and analyze their different characteristics in terms of performance and energy consumption.
- We test these policies on larger ring-based photonic networks, showing their general applicability to hybrid photonic-electronic networks where the optical resources are shared or partially shared between the cores.

The rest of this paper is organized as follows. Section 2 discusses the photonic architectures considered in the paper. Section 3 describes the management policies and their strong and weak points. Section 4 evaluates them from both performance and energy-consumption points of view. A survey of related work can be found in Section 5, and, finally, Section 6 concludes.

2. BACKGROUND

This section gives the necessary background on photonics and on the network architectures selected to evaluate our policies.

2.1. Data transmission with silicon photonics

The basic elements necessary to build a photonic network are waveguides, light sources, modulators, and detectors. *Silicon waveguides* are on-chip channels that carry light modulated to convey information, possibly enabling communication across the chip over longer distances, at higher bitrates and with lower losses than electrical wires, resulting in lower delays and avoiding the need for retransmissions. A high refractive index difference between the waveguide core and the cladding ensures that light is driven efficiently, preventing losses and crosstalk between close waveguides. As a *light source*, either on-chip or off-chip lasers are used to inject light into the waveguide. Wavelength division multiplexing (WDM) consists of using light with different frequencies or wavelengths to transmit information in several channels simultaneously, with 1 bit of width for each channel, corresponding to a particular wavelength or *carrier*. *Microring resonators* are made of looped optical waveguides [16] that can be used as modulators and filters. Very small microring resonators are possible with silicon (e.g., with a radius of 1.5 μm [17]), playing a major role in the success of on-chip photonics.

Figure 1 shows all the elements involved in the process of transmitting data with photonics. A light source injects all the wavelengths that carry data (two in this case) into the waveguide. There are specific microrings for each wavelength at the network endpoints, which are used to modulate and to detect information encoded in the light signal. The operation of these microrings is based on resonance, which occurs when the optical path length of the microring resonator is exactly a whole multiple of the wavelength of an optical signal that is being transmitted on the adjacent waveguide.

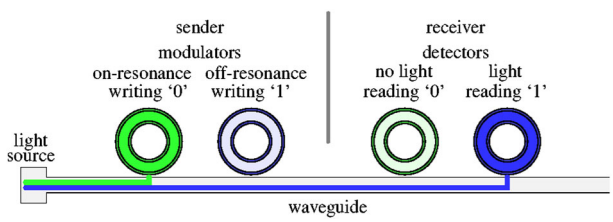


Figure 1. Basics of photonic data transmission with just two wavelengths (blue and green). Each microring resonator is associated with one wavelength that makes the microring be on resonance. Four microring resonators appear in the figure, a modulator and a detector for each wavelength. A modulator on resonance extracts the light of its associated wavelength, causing the corresponding detector to read a '0' value (see the green wavelength). A modulator electrically made be off-resonance lets light pass, and the corresponding detector reads a '1' value as a result (see the blue wavelength).

DOI: 10.1002/cpe

The resonant frequency of a microring can be adjusted modulating a bias voltage for heating or for injecting carriers into the silicon of the microring. This way, the microring can be finely tuned to the required frequency or put out of resonance when needed.

When being used for sending information, microring resonators are electrically actuated to modulate the light signal. For instance, in case of an *on-off* modulation strategy, an on-resonance microring drops the light circulating in the waveguide corresponding to its resonant wavelength, obtaining the injection of the '0' value. Putting the microring out of resonance allows the light to proceed through the waveguide, resulting in the injection of a '1' value. The desired stream of '0's and '1's for the transmission is achieved by bringing the ring in and out of resonance appropriately. On the other hand, when acting as a detector, the microring is on resonance, which causes any light corresponding to its resonant wavelength passing through the adjacent waveguide to be coupled into the microring. Notice that this detection procedure removes the signal from the waveguide. In order to translate the optical signal into the electronic domain, typically a Germanium doped photodiode, embedded in the microring, detects a logic '1' each time that light is coupled into the microring and detects a logic '0' when no light is coupled into the microring.

2.2. *Ring-based photonic networks*

Because of their simplicity and flexibility, optical topologies based on open or closed waveguides that implement logical *rings* [6–8, 12] are likely to make their way into commercial machines before more complex photonic architectures like those that use photonic switches, *passive* or *active* (i.e., dynamically reconfigurable) [9, 10, 18, 19].

Passive switches always let one or more wavelengths go through the switch without turning and divert one or more wavelengths to a different photonic output port [18, 19]. This can enable a passive-routing interconnection by associating the wavelengths to origin-destination node pairs. These approaches need to employ a significant number of optical switches and incur many waveguide crossings, which can introduce significant optical attenuations, but are able to deliver dedicated optical channels between each core pair. However, because of intrinsic technological limitations of optical switches and to the specific features of coherency traffic [7], these passive networks cannot deliver great optical parallelism per source-destination pair, at a given overall on-chip optical resource provisioning. On the other hand, the use of active photonic switches [9, 10] needs a photonic-circuit establishment mechanism to configure the microring resonators through a supporting electrical network, before transmission. This is essentially due to the incapacity to implement routing within the optical domain. The circuit establishment overhead makes these photonic topologies less attractive for cache coherence-based systems in which communications typically consist of quick transmissions of small packets (around 8 or 72 bytes typically [20]). In this case, the overhead of establishing the circuit in the mesh would largely outweigh the benefits of sending the packet through the photonic ring.

Figure 2 exemplifies a ring-shaped waveguide in a hybrid photonic-electrical NoC for a 4×4 CMP. Even simple topologies expose various design choices to be made [21]. For example, each

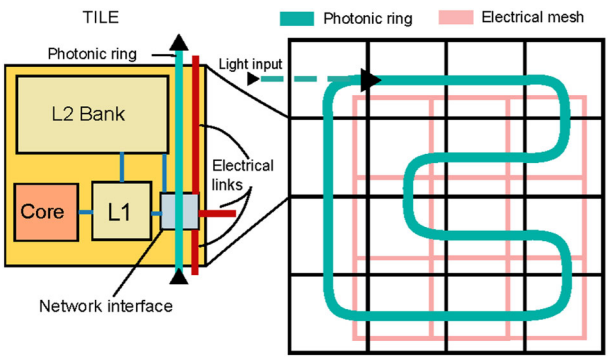


Figure 2. Hybrid photonic-electronic network-on-chip on a 16-core tiled CMP. Every tile can read from or write in the photonic ring.

DOI: 10.1002/cpe

origin-destination pair can be assigned to a different set of wavelengths, preventing the need for arbitration. However, this would limit the bandwidth for single transmissions noticeably. Other designs provide flexibility but require some arbitration. A Single Writer Multiple Reader (SWMR) [12, 22] configuration allocates a different set of wavelengths for each writer, which uses a destination selection mechanism to make the desired receiver turn on its photodetectors to read the data. In Multiple Writer Single Reader (MWSR) [6], each receiver reads different wavelengths, and arbitration is needed on the writer's side to avoid collisions between writers. Multiple Writer Multiple Reader (MWMR) [7] is the most flexible, allowing a single transmission to use all the data wavelengths of the ring, but requires both arbitration in the writer's side and destination selection as well as more ring modulators and photodetectors for every destination to read and write every wavelength, which increases area and power consumption.

2.3. *Case study photonic networks*

Here, we describe three notable networks that use ring-based photonic topologies, which have been recently proposed, one for each ring arbitration policy. Table I describes the characteristics of these networks, with all of them using dense wavelength division multiplexing (DWDM) where up to 64 wavelengths are transmitted through a single waveguide. Later, the ability of our policies will be tested to exploit hybrid NoCs based on these photonic networks.

2.3.1. FlexiShare (MWMR). FlexiShare [7] is an MWMR photonic ring proposed for a 64-core CMP. It introduces token stream arbitration to increase network utilization. FlexiShare was evaluated by their authors with varying radix (8, 16, 32) and concentration (8, 4, 2) values. Also, different numbers of channels, each with a 64-bit datawidth, were tested. Arbitrating these channels is not trivial [23], and FlexiShare assumed a round-robin channel selection. The values in Table I assume eight channels.

2.3.2. Corona (MWSR). Corona [6] is a 256-core CMP containing a ring-based MWSR photonic network to interconnect 64 four-core clusters. Each of the 64 endpoints receives data through a dedicated 256-bit datapath that comprises four photonic waveguides, and senders compete for the right to use the channel. The resources needed by Corona are much higher than those required by the 64-core FlexiShare designs, and it solves the multiple-channel arbitration problem by using dedicated channels for receivers, at the cost of wasting bandwidth under unbalanced traffic.

Table I. Feature comparison between case study networks-on-chip (NoCs). Values with (*) are estimations based on the available information.

	System characteristics			Electronic features	
Solution	Technology	Cores	Notes	Concentration	N_{links}
FlexiShare	22 nm	64	Eight 512-bit channels	8–2	0
Corona	16 nm	256	Photonics to DRAM	4	0
Firefly	45 nm	256	Hybrid ph/el NoC	4	80
	Optical features				
Solution	N_{endp}	N_{waveg}	$N_{micror.\ res.}$	Access scheme	Phit size
FlexiShare	8–32	130–138*	113–1052 K*	MWMR	512
Corona	64	388	1056 K	MWSR	256
Firefly	64	320*	130 K*	SWMR	256

MWMR, multiple writer multiple reader; MWSR, multiple writer single reader; SWMR, single writer multiple reader.

DOI: 10.1002/cpe

2.3.3. Firefly (SWMR). Firefly [12] is a hybrid photonic-electronic network using photonic rings also for 256-core CMPs. In addition to using a concentration of four cores per endpoint, like Corona, Firefly's design groups endpoints in eight clusters. An electrical mesh per cluster carries intra-cluster traffic benefiting from the high bandwidth of electrical links in short-distance transmissions. An SWMR photonic ring is used for inter-cluster traffic to enable fast long-distance communication, with a dedicated 256-bit channel per writer. The channel connects the writer to just one endpoint per cluster to save photonic resources compared to Corona, where channels connect all 64 endpoints. This results in rings being efficiently used for long-distance communication (inter-cluster) without suffering the burden of short-distance transmissions (intra-cluster) that would increase packet serialization. This also enables the removal of electrical links between clusters, providing static energy and area savings compared to a full mesh.

2.4. Arbitration and pipelined transmission

Several arbitration mechanisms are possible for the photonic rings just discussed. For MWMR and MWSR rings, we use a simple token-passing arbitration mechanism because of its simplicity and its fairness. Using this technique, an emitter reads the token wavelength, and when a light pulse is detected (and therefore extracted from the waveguide), the token has been acquired. We allow each emitter to send one message, and then the emitter has to inject the token again in the waveguide. The main problem of this simple token passing mechanism is the underutilization of the photonic ring that may appear under certain conditions such as a single emitter that has to relinquish the token periodically and wait for its arrival after circling the ring before transmitting again, wasting potential transmission slots in the ring. However, results show that a very high utilization is achieved with the single token ring mechanism across the PARSEC benchmark suite. Furthermore, we implement an optimization of such logical token scheme that pipelines the token acquisition, destination selection (needed in MWMR), token release, and data transmission; and leverages signal propagation delay and separated wavelengths for token and data signals. For instance, we assume that token release by a transmitter can occur up to two optical cycles *in advance* of actual data transmission (or the transmission of the last flit of multiple-flit messages). This indeed allows our token scheme to guarantee practically full-bandwidth utilization. Lastly, our setup employs a cache coherency protocol, which uses 1-flit long control messages (8 byte = 64 bit) and 9-flit long data messages (72 byte = 572 bit). Therefore, it is desirable to adopt a token strategy like this for maintaining data transmission latency low through the intrinsic ability of keeping the channel occupied for multi-flit transmissions. Nevertheless, more complex arbitration techniques can potentially give a marginal improvement on the utilization of the ring. In any case, evaluating different arbitration mechanisms is out of the scope of this paper, and the proposed policies are equally applicable along with more complex arbitration mechanisms. For SWMR rings, the arbitration mechanism is not necessary, as each emitter uses a dedicated datapath.

For the sake of fairness in the evaluation, we assume the same pipelined packet transmission for all networks. Between token acquisition and data transmission, a lapse of three ring-cycles takes place in which the activation of the destination's photonic receivers is performed by means of a light pulse on four wavelengths, indicating the identity of the destination. This is not strictly necessary for MWSR rings as the wavelengths used for the transmission are associated to one destination, but we use it because the destination receivers may be turned off to save energy in the absence of transmissions. The token is released before the real transmission takes place to enable the potential use of all slots. All of the latencies involved in the arbitration have been modelled in the evaluation.

3. DYNAMIC MANAGEMENT POLICIES FOR HYBRID NOCS

This section presents a set of novel policies to manage hybrid networks comprising a ring-based photonic sub-network and an electrical sub-network such as a mesh. They decide which sub-network

DOI: 10.1002/cpe

to use for each message using real-time information based on the following parameters: message size, photonic ring availability, and distance between endpoints.

3.1. SIZE: Message size

Our first criterion to decide which subnetwork to use is extremely simple, taking into consideration just the size of the message. There are typically two different kinds of messages in a cache-coherent CMP: control messages and data messages, with typical respective sizes of 8 and 72 bytes (the 64-byte difference is accounted by the data block) [20]. Transmitting a data message makes the photonic ring unavailable for much longer than a control message, potentially increasing the latency of other messages.

For instance, the near-future network evaluated in Section 4 comprises a high-performance waveguide with 64 data wavelengths that enables the transmission of up to 8 bytes per ring-cycle. Therefore, sending control and data messages takes one and nine ring-cycles, respectively. In nine ring-cycles, nine short messages (one per ring-cycle) could be sent, greatly benefiting from the low latency of the ring. That is, sending long data messages increases the chances of creating long message queuing times due to serialization and decreases the opportunities to accelerate many other messages.

In addition, short messages account for just a small percentage of the overall traffic in bytes because of their small size, although they account for most of the messages injected. Thus, their acceleration provides a large performance gain with small bandwidth usage.

All of this makes it interesting to use a simple policy, which we call *SIZE*, that only sends through the photonic ring those messages of small size (control). Notice that the opposite policy (sending data messages on the photonic ring) would be prone to serialization, resulting in high queuing times.

However, this policy has some shortcomings as it cannot adapt to the burst nature of traffic in parallel applications. Under low traffic loads, the fixed-message-size criteria can under-utilize the photonic resources, especially in large photonic NoCs, wasting the opportunities to accelerate and reduce the energy consumption of other messages. On the other extreme, under high traffic loads or traffic bursts, this policy can completely lose the latency benefit of the photonic NoC when too many short messages contend at a given moment, causing their serialization especially if there are limited photonic resources.

3.2. AVAIL: Ring availability

This policy sends a message (control or data) through the photonic ring only if the ring is readily available when the transmission is attempted or before a parametrized wait time passes. Hence, the electrical mesh is used for those messages that find the photonic ring busy. This adjusts dynamically the traffic injected in each sub-network, preventing the shortcomings of *SIZE*.

Because we have to acquire the token before transmitting, we must wait for the token round-trip time (two processor cycles in the NoC evaluated in Section 4.3) before knowing whether the ring is busy or not. If the token is acquired, then the message is sent through the ring. If it is not, the ring is busy, and the message is sent through the mesh (after having waited unfruitfully during two processor cycles).

With this policy, we make sure that messages are never serialized waiting for the photonic ring under high traffic load scenarios (they use the mesh after failing one attempt of token acquisition), and that every message has a chance of using the photonic ring. Also, for low traffic loads, we prevent messages from being sent through the mesh while the photonic ring is free, hence increasing ring utilization and reducing overall energy consumption.

However, with this policy, data messages can monopolize the photonic ring for long times, forcing short messages to be sent through the mesh. Moreover, when concurrent transmissions from different nodes take place, only one of them gains access to the photonic ring while the others have to use the mesh even when it would be more beneficial overall that they waited a few extra cycles and sent their messages through the photonic ring after the first node. To account for this, we explore several token-wait delays in the evaluation section under the name *AVAIL*.

DOI: 10.1002/cpe

3.3. DDA: Distance Dependent Availability

The energy consumption and latency of a message transmission through an electrical mesh varies depending on the distance between the origin and the destination. In a 4×4-CMP, a message between opposite corners of the chip requires six intermediate routing operations and six retransmissions through inter-tile electrical links. Transmitting the same message between adjacent tiles requires just one routing operation and one transmission on the link connecting the tiles. Therefore, communication between distant destinations requires much more energy and takes longer.

Moreover, this means that not every core is affected by network latency in the same way with an electronic network. Cores in the corners and borders of the chip suffer from longer average network distances and latencies than those in the center [24], making threads running on them execute more slowly than those running on central cores and harming parallel application performance, as slower cores limit the overall execution speed.

Electrical links can provide high-bandwidth short-range connectivity efficiently. For instance, a 128-wire electrical link operating at 4 GHz can provide a bandwidth of 64 GB/s in one direction between neighbor tiles in a CMP, which is comparable to a typical photonic waveguide, which can manage 64 independent data wavelengths, providing a bandwidth of 80 GB/s when operating at 10 GHz. We want to exploit this characteristic of electrical meshes with our policies.

Table II shows the percentage of messages and link traversals caused by communications between cores at different distances for a uniform random distribution of accesses to a NUCA LLC in a 16-core CMP, which matches our observations on the PARSEC benchmarks. Transmissions between neighbor nodes (1 hop) account for 20% of messages but only generate 7.5% of link traversals, while 5-hop transmissions account for 6.7% of messages but generate a significant 12.5% of link traversals. Because the photonic ring is almost insensitive to distance, a smart use of it would be sending through it those messages that would incur the most energy consumption and latency if transmitted through the mesh (that is, messages between distant endpoints) and then use the mesh for short-distance transmissions. This would also make cores far from the center of the chip benefit greatly, making them catch up with center cores, boosting performance even further.

For exploiting photonics efficiently, cores far from the center of the chip should make more use of the photonic ring (e.g., only the cores in the opposite corners are 6-hops away from each other, and messages between them should go through the photonic ring). This way, the effective network latency differences between cores will shrink, reducing the overhead of synchronization operations on the execution time and considerably boosting performance.

To leverage this, we have developed a heuristic policy called distance dependent availability (DDA), which combines the benefits of preventing long waits for the photonic ring and favoring its use for distant endpoint communications. We achieve this mix of goals by allowing a different token-wait time for each particular message that is proportional to the theoretical advantages of using the photonic ring over using the mesh. To calculate this advantage of the ring, we use the best case theoretical latency of transmitting each message in the ring (l_p) and in the mesh (l_m) in the absence of other transmissions. In our setup, $l_p = 2$ or $l_p = 5$ processor cycles in case of a control (8-byte) or data (72-byte) message, respectively and, correspondingly, $l_m = 5/hop$ processor cycles for control messages and $l_m = 5/hop + 8$ for a data message. Every message is, at first, considered for sending through the photonic ring. If the ring is found idle, the message is sent.

Table II. Hops and link traversals in a 4 × 4 mesh for messages to access the last-level cache that leave the tiles.

Distance (in hops)	Messages	Link traversals	Aggregate messages	Aggregate link traversals
1	20.0%	7.5%	20.0%	7.5%
2	28.3%	21.3%	48.3%	28.8%
3	26.7%	30.0%	75.0%	58.8%
4	16.7%	25.0%	91.7%	83.8%
5	6.7%	12.5%	98.3%	96.3%
6	1.7%	3.8%	100.0%	100.0%

DOI: 10.1002/cpe

Otherwise, the message waits for $(l_m - l_p) \times th$ cycles, where *th* is a configurable threshold with values between 0 (no wait) and 1 (wait as long as there is any potential benefit in using the ring), before sending the message through the mesh. Small values of *th* avoid serialization of messages, while large values increase ring utilization. In any case, messages involving distant endpoints wait longer, hence acquiring the token and using the photonic ring more often. In the evaluation section, we consider several values for *th* to explore possible trade-offs.

In order to account for the different sizes of the messages and exploit the potential benefits, we give differentiated treatment to them in the photonic ring in two additional heuristic policies explained in the succeeding text that add message size to the variables considered for photonic ring management.

Control DDA (CDDA) consists of using DDA for control messages and AVAIL for data messages. This policy tries to obtain an average low message latency by transmitting many short messages, while keeping a high utilization of the ring by sending data messages when it is otherwise idle.

Multi-threshold DDA (MTDDA) uses DDA for both control and data messages, but different thresholds are used for each message type. A longer threshold is used for control messages to prioritize their transmission in the ring. In this case, we give more importance to the ring utilization than in CDDA, because data messages are now more likely to be transmitted.

3.3.1. Dynamic Thresholds. The burst nature of traffic can make dynamic thresholds desirable to avoid unnecessarily long waiting times. Under low traffic loads, a high threshold increases the ring utilization without incurring severe message serialization. If the traffic load goes up, the threshold can go down to reduce the token waiting times and still keep a high utilization of the photonic ring.

We have explored several dynamic threshold designs, also differentiating by message size, but only marginal improvements were observed in our experiments. Traffic patterns change at a very fine granularity, so the adaptive thresholds provided only small benefits in terms of execution time or energy consumption when compared to the simpler policies proposed so far in this paper. We consider that these results are not significant enough to be included in the evaluation section of this paper. Nevertheless, more elaborated dynamic-threshold mechanisms capable of capturing the behavior of the network would be more interesting, especially if they could predict future traffic trends and adapt in advance.

3.4. Photonic-electronic interface

An adequate interface between the photonic and electronic sub-networks is required to apply our policies. This interface adds little complexity over the one used in previous works (e.g., Firefly [12]).

In each tile of the CMP, one *pre-photonic buffer* interfaces each traffic source (e.g., L1 or L2 caches) with its associated external port to the electrical mesh router. Upon injection, those network messages that are candidate for photonic transmission (e.g., control messages in SIZE) are first stored in the corresponding pre-photonic buffer to wait for the photonic ring. These messages are eventually sent either through the photonic ring or transferred to the corresponding external input port to the router for electrical transmission.

At any given moment, only one of the pre-photonic buffers of the node can be *active*, meaning that the message at its head is the one being considered for photonic transmission. As long as any pre-photonic buffer is active, the token acquisition photodetector of the tile remains on (it is off otherwise). As soon as the token is acquired, the active message is transmitted through the photonic ring, and then the token is re-injected.

In addition, when a message enters a pre-photonic buffer, a countdown timer associated to the entry storing the message is set to the appropriate waiting time (depending on the policy). If the timer reaches zero, photonic transmission is ruled out, and the message enters the electrical router's external input port. Notice that these timers are not needed by SIZE.

To ensure photonic transmission fairness between the tile's traffic sources, when the active message is photonically transmitted or when its associated counter reaches zero, a round-robin algorithm activates the next pre-photonic buffer containing messages, if any.

DOI: 10.1002/cpe

4. EVALUATION

In this section, we discuss the performance of the management policies proposed for photonic-electronic hybrid networks. We have performed two sets of experiments to analyze the policies of Section 3. First, we test the policies using detailed full-system simulation on a photonic ring based on FlexiShare [7] that could be implemented in the near future. Then, we test the policies on larger networks based on Corona [6] and Firefly [12] by means of synthetic traffic-based simulations. See Section 2.3 for a brief description of these networks. Table III shows the adapted NoCs used for our tests and our estimated target date of availability for these networks. The last column of the table shows the electronic resources (links) that must be added with respect to the original proposal to create the full mesh assumed by our policies. Also, by testing the policies in NoC of several sizes, we show their general applicability to exploit hybrid networks.

4.1. Evaluation methodology for 16-endpoint NoCs

We scale down FlexiShare to just one waveguide, resulting in an affordable design suitable for near future commercial CMPs that requires just around 2000 microring resonators. Figure 2 shows our base hybrid photonic-electrical NoC in a 4×4 CMP. We consider a realistic five ring-cycle full-ring traversal time at 10 GHz for light pulses (i.e., two processor cycles at 4 GHz). All of the data wavelengths of the ring can be simultaneously used by one emitter to communicate with one receiver, and we limit the number of concurrent transmissions in the ring to just one. In all, this FlexiShare-like configuration requires 65 wavelengths for its operation. During destination selection, four wavelengths identify the receiver, and one wavelength indicates the size of the message to transmit (1 bit is enough to encode the size, because only two sizes exist corresponding to control and data messages). Sixty-four wavelengths are used for data transmission. One extra wavelength is needed to circulate the single-bit token in which the arbitration mechanism is based.

As for flow control, although the receivers have enough buffering resources to accommodate the traffic in the common case, buffer overflow may appear occasionally. In that case, a NACK signal is sent by the receiver through the data wavelength in the complementary ring portion to the transmission (hence, closing the circle without disturbing any other photonic transmission), and it is read by the transmitter that backs down, releases the token (if it had not been released yet), and repeats the same transmission procedure again after some time. This flow control mechanism is rarely needed and has little impact in the overall performance. More complex mechanisms can be used, but their evaluation is out of the scope of this paper.

This flexible MWMR configuration allows for higher utilization of resources and faster single transmissions than MWSR and SWMR rings, especially under unbalanced traffic, as shown in [7] for the SPLASH-2 benchmarks. We have observed that the same holds true for the PARSEC benchmark

Table III. Features of adapted case study network-on-chips in which the policies are applied.

	System characteristics			Electronic features		
Solution	Technology	Year	Cores	Concentration	N_{links}	N_{links}^{extra}
FlexiShare	35 nm	2014	16	1	0	24
Corona	16 nm	2018	256	4	0	112
Firefly	16 nm	2018	256	4	80	32
	Optical datapath features					
Solution	N_{endp}	N_{waveg}	$N_{micror.\,res.}$	Access scheme	Phit size	
FlexiShare	16	1	2 K	MWMR	64	
Corona	64	256	1024 K	MWSR	256	
Firefly	64	256	128 K	SWMR	256	

MWMR, multiple writer multiple reader; MWSR, multiple writer single reader; SWMR, single writer multiple reader.

DOI: 10.1002/cpe

suite. Even so, the MWMR ring is unable to carry all traffic in a 16-core CMP by itself, and we have measured that the execution time for the PARSEC benchmarks is on average 2.2 times larger when using it than when using the electrical mesh. Thus, ideally, our policies should use such photonic ring selectively as a sort of *accelerator* for the mesh in this scenario.

4.1.1. Simulated CMP and benchmarks used. The GEM5 simulator [25] was used to perform these tests. The common characteristics of the 16-core simulated CMP can be found in Table IV. The L2 cache uses a Non-Uniform Cache Architecture (NUCA) design and a directory-based MOESI protocol enforces coherence between the private L1 caches. We have assumed a high-performance 4×4 electrical mesh running at 4 GHz, consisting on bi-directional 1-cycle latency 128-wire links and four-stage pipelined routers. We consider an optimized router architecture in which the destination router requires just 1 cycle to deliver the message to the appropriate output buffer, instead of 4 cycles. Under these assumptions, a message transmission between two adjacent routers requires just 6 cycles for the first flit to go from the initial buffer in the mesh to the final buffer (one 1-cycle link traversal, one 4-cycle router traversal, and a 1-cycle router traversal), and between the most distant routers, it requires 31 cycles (six 1-cycle link traversals, six 4-cycle router traversals, and one 1-cycle router traversal).

On its part, transmitting on the 10 GHz MWMR photonic ring when it is idle requires to acquire the free token (up to five ring cycles), activate the destination's receivers (three ring cycles) and then reach the destination with the first photonic pulse (up to five ring cycles). In the case of control messages, this is enough to transmit the 8-byte message. In the case of data messages, a second photonic pulse carries the 8-byte requested word (the first pulse contains an 8-byte header). The rest of the data block is transmitted in consecutive photonic pulses.

Table IV. Simulated machine.

Processors	16 alpha cores @ 4 GHz, 2-ways, in-order
L1 Cache	Split I&D. Size: 16 KB, 4-ways, 64 bytes/block Access latency: 1 cycle MOESI coherence protocol (directory cache in L2 cache)
L2 Cache	Size: 1 MB per bank. 16 MB total (NUCA) 8-ways, 64 bytes/block Access latency: 15 cycles
RAM	4 GB DDR2 DRAM 16 3D-stacked memory controllers
Interconnection - electronic	4 GHz, 2D mesh: 4×4 16-byte links. Latency: 1 processor-cycle/link 4-processor-cycle pipelined routers Flit size: 16 bytes Control/data packet size: 8/72 bytes (1/5 flits) Dynamic energy (1-hop switch+link): 282 pJ/flit = 17.625 pJ/bit Static power (switch+link): 52.7 mW
Interconnection - photonic	10 GHz MWMR Photonic Ring. 3D-stacked 65 wavelengths Latency: two processor-cycle round-trip time Two processor-cycle minimum transmission time on idle (no token wait, closest node) Six processor-cycle maximum transmission time on idle (round-trip time token wait, furthest node) Flit size: 8 bytes. Control/data packet size: 8/72 bytes (1/9 flits) Dynamic energy: 0.41 pJ/bit Static power (laser+microrings): 318 mW

DOI: 10.1002/cpe

In the most favorable case (no wait for the token and a transmission to the closest node), an idle photonic ring provides a two-processor-cycle transmission latency (rounded up) for control messages or for the requested word of data messages. In the most unfavorable case (round-trip time wait for the token and transmission to the farthest node), this latency increases to six processor-cycles. In comparison, the idle 4×4 electrical mesh requires up to 31 processor cycles between the most distant destinations. Time and power parameters of the electronic NoC are derived from Orion 2.0 [26] using a 32-nm silicon process. We consider state-of-the-art optical devices [27] and their behavior in our reference architecture. In particular, the table highlights the dynamic and static components of the network elements (switches and links) used to calculate the electronic baseline network consumption. For the optical network, the dynamic component is mainly determined by modulators and photodetectors while the static quote comprises microring thermal tuning and on-chip dissipated laser power, in turn dictated by worst path insertion loss, laser efficiency, and photodetector sensitivity [28].

We have used the PARSEC benchmark suite with the medium-sized working sets to perform this study. We evaluate all the policies described in Section 3 with several configurations each when appropriate. Table V shows the policies evaluated with the specific parameters used.

4.2. Evaluation methodology for 64-endpoint NoCs

We also evaluated our policies on larger NoCs. For this, Corona [6] and Firefly [12] were chosen as good examples of future NoCs for 256 cores (four cores per endpoint). The NoC parameters were set to match those of the original papers unless stated otherwise. Table VI describes the five synthetic traffic patterns used in the tests, which are inspired by [29]. Of these, uniform traffic is the most similar to the one observed in real applications on a NUCA cache.

Table V. Evaluated policies.

Configuration	Messages on photonic ring	Messages on mesh
Mesh	None	All
SIZE	control messages (8 bytes)	data messages (72 bytes)
AVAIL-2	token acquired within 2 cycles	other messages (2-cycle delay)
AVAIL-6	token acquired within 6 cycles	other messages (6-cycle delay)
AVAIL-10	token acquired within 10 cycles	other messages (10-cycle delay)
DDA-25	token acquired within $(l_m - l_p) \times 0.25$ cycles	other messages ($(l_m - l_p) \times 0.25$ delay)
DDA-50	token acquired within $(l_m - l_p) \times 0.50$ cycles	other messages ($(l_m - l_p) \times 0.50$ delay)
DDA-75	token acquired within $(l_m - l_p) \times 0.75$ cycles	other messages ($(l_m - l_p) \times 0.75$ delay)
CDDA-25	control if token acquired within $(l_m - l_p) \times 0.25$ cycles data if token acquired within 2 cycles	other messages ($(l_m - l_p) \times 0.25$-cycle delay for control, 2-cycle delay for data)
CDDA-50	control if token acquired within $(l_m - l_p) \times 0.50$ cycles data if token acquired within 2 cycles	other messages ($(l_m - l_p) \times 0.50$-cycle delay for control, 2-cycle delay for data)
CDDA-75	control if token acquired within $(l_m - l_p) \times 0.75$ cycles data if token acquired within 2 cycles	other messages ($(l_m - l_p) \times 0.75$-cycle delay for control, 2-cycle delay for data)
MTDDA-60-40	control if token acquired within $(l_m - l_p) \times 0.60$ cycles, data if token acquired within $(l_m - l_p) \times 0.40$ cycles	other messages ($(l_m - l_p) \times 0.60$-cycle delay for control, $(l_m - l_p) \times 0.40$-cycle delay for data)
MTDDA-75-25	control if token acquired within $(l_m - l_p) \times 0.75$ cycles, data if token acquired within $(l_m - l_p) \times 0.25$ cycles	other messages ($(l_m - l_p) \times 0.75$-cycle delay for control, $(l_m - l_p) \times 0.25$-cycle delay for data)

l_m, idle mesh latency; l_p, idle photonic ring latency.
All latencies in processor cycles at 4 GHz.

DOI: 10.1002/cpe

Table VI. Synthetic traffic patterns.

Traffic name	Details
Uniform	Uniform random traffic
Transpose	$(i, j) \Rightarrow (j, i)$
Bitcomp	dest id = bit-wise-not(src id)
Neighbor	Randomly send to one of the source's neighbors
Tornado	$(i, j) \Rightarrow ((i + \lfloor X/2 \rfloor - 1) \mod X, (j + \lfloor Y/2 \rfloor - 1) \mod Y)$

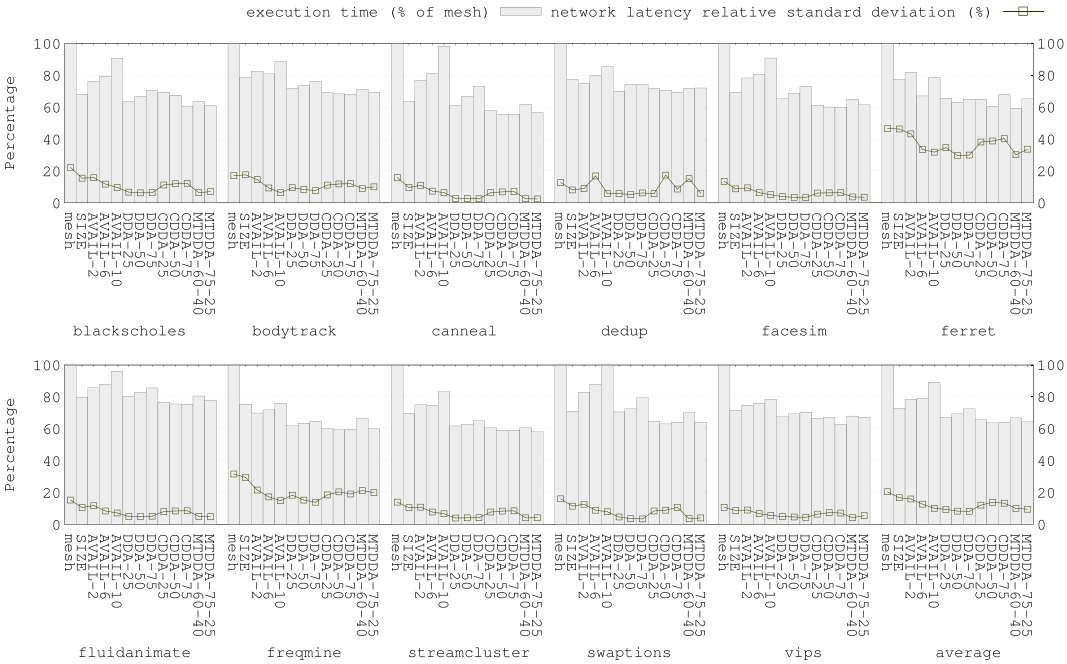


Figure 3. Execution time. Normalized to electronic mesh.

The cumulative injection rates for the four concentrated processors, in packets per cycle, are used as the load metric. Short (64-bit) and long (576-bit) packets were injected randomly. Four-cycle routers were used in the mesh.

We do not include the results for FlexiShare in this evaluation for a number of reasons. First, the conclusions regarding Flexishare and the policies are similar to those of the 16-endpoint scenario, not unveiling any new significant information. FlexiShare shows better performance compared to Corona and Firefly for the same amount of photonic resources, which is explained by the superior flexibility of the MWMR arbitration, but such comparison of photonic network layouts is out of the scope of this paper. Also, FlexiShare was originally proposed with layouts from eight to 32 endpoints [7], and the particular 64 endpoint configuration employed by us may not be necessarily comparable to Corona and Firefly in terms of power. Finally, we are working on novel policies to additionally take into account the existence of multiple MWMR channels to carry out communication more efficiently than with FlexiShare's random selection policy.

4.3. Results for 16-endpoint hybrid NoC

Figure 3 shows, for each policy, the execution time and the relative standard deviation of the network latency suffered by the cores. Figure 4 shows the average latencies for message transmissions in the electrical mesh, in the photonic ring, and overall. Although the frequencies of these networks are different (4 GHz for the electrical and 10 GHz for the photonic), all the data are plotted at 4 GHz. We also show the theoretical latency gain for the messages that are finally sent through the pho-

DOI: 10.1002/cpe

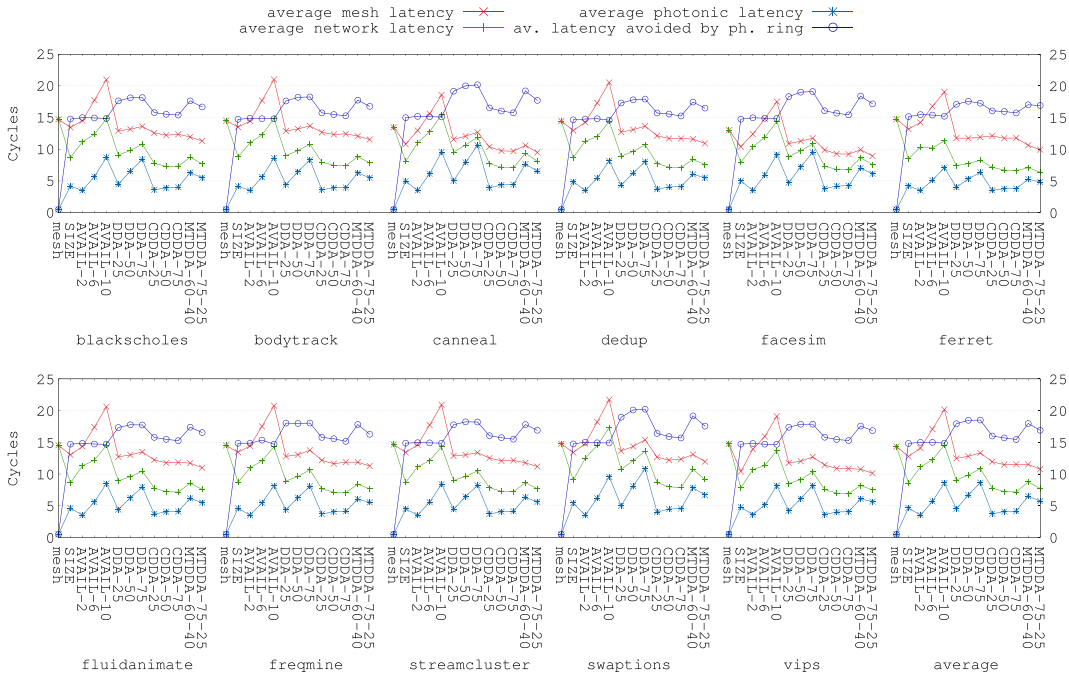


Figure 4. Network latency. Normalized to electronic mesh.

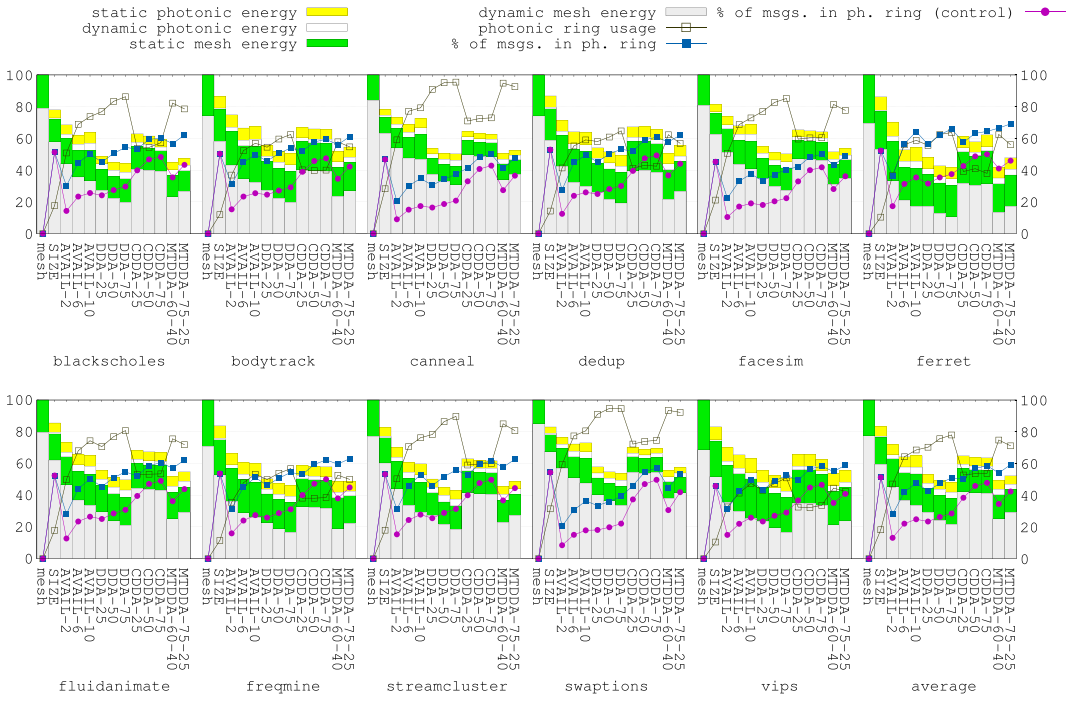


Figure 5. Network energy consumption (normalized to electronic mesh) and photonic ring usage.

tonic network. Figure 5 shows the energy consumption of the electrical mesh and the photonic ring, along with the photonic network usage and the fraction of messages sent through it. A higher percentage of messages does not imply a higher utilization of the ring, because two different message sizes exist.

DOI: 10.1002/cpe

In general, all of the photonic management policies improve performance compared to the baseline electrical mesh. Also, the trends shown by each policy remain stable across all benchmarks.

The SIZE policy reduces execution time by 27% with respect to the baseline mesh thanks to a reduction in the latency of control messages. The latency gets reduced by 40% on average. In general, the wait time for acquiring the ring is low (1.2 cycles) thanks to the small size of the messages transmitted, avoiding serialization. However, the ring is underutilized because many times there are no control messages to transmit. Nevertheless, a 17% reduction of the energy consumption of the network is achieved with respect to the base architecture.

The execution time of AVAIL, which only sends messages when the token can be acquired before a number of cycles pass, is noticeably higher than SIZE's, as AVAIL-2 reduces execution time by just 21% compared to the baseline. The latency reduction is also lower. The reason is the larger average size of messages transmitted through the ring (control and data in AVAIL, only control in SIZE) that causes fewer messages to be accelerated compared to SIZE (each data message sent prevents sending up to nine control messages). As the number of wait cycles increases—AVAIL-6 and AVAIL-10—the average network latency increases too, resulting in average reductions of just 20% and 11% in time over the baseline. However, the sending of data increases the usage of the ring (48% in AVAIL-2), increasing the energy reduction (28% in AVAIL-2). In addition, increasing the wait cycles also increases the usage of the ring (65% and 69% for AVAIL-6 and AVAIL-10), which results in a noticeable reduction of the energy consumption of the network (36% for both). We can say that AVAIL prioritizes energy reduction over execution time when compared to SIZE. Also, the wait threshold provides a way to tune between lower execution time and lower energy consumption.

Distance-based policies make more efficient use of photonics and improve the performance results of SIZE and AVAIL. DDA-25 reduces execution time by 33%. As the policy threshold goes up, the execution time average benefit is reduced to 30% (DDA-50) and 27% (DDA-75) over the baseline. This performance drop is caused by the increase in the average wait times to transmit when using higher thresholds. DDA provides 37%, 31%, and 24% lower network latencies on average for 25%, 50%, and 75% thresholds, respectively, because the photonic ring is preferentially used to send long-distance messages, while the electrical mesh is now mainly used for short-distance messages. DDA achieves the highest reductions in energy because its photonic ring usage is the highest, and the ring is used for messages between distant endpoints, reducing the need for retransmissions in the mesh. Similarly to AVAIL, DDA's thresholds are useful to trade-off between speed and energy consumption.

CDDA, which uses DDA for control messages and AVAIL for data messages, provides the highest reduction in execution time. CDDA-25 reduces execution time by 33%, and this value increases to 36% for CDDA-50 and CDDA-75. In CDDA, the distance-dependent thresholds keep high the mesh latency avoided by the ring, although lower than in DDA because now most control messages use the photonic ring, including messages between close endpoints. The latency reduction is the highest of any policy thanks to the higher number of messages accelerated. The energy consumption of CDDA is higher than DDA (reductions of 35%, 36% and 36%) due to the lower usage of the ring (53%, 54%, and 55%).

Finally, MTDDA, which uses different thresholds for control and data messages, performs close to CDDA in execution speed and network latency, with 33% and 35% lower execution times than the baseline for MTDDA-60-40 and MTDDA-75-25, respectively. These policies allow some wait for data messages, based on distance, in order to achieve a balance between execution time speedup and photonic ring utilization. By using different thresholds for control and data, we can still prioritize the sending of short messages in order to reduce execution time while retaining the ability to achieve a high utilization of the photonic ring with data messages. This results in a good trade-off between DDA and CDDA. We believe that MTDDA is the most versatile policy, providing good results in both execution time (almost matching CDDA) and energy consumption (close to DDA).

As mentioned in Section 3.3, cores in the center of the chip suffer less latency when accessing the LLC. Figure 6 shows the overall absolute latency suffered by each core for the electrical mesh and for MTDDA-75-25 in each benchmark. The pattern observed in the mesh matches closely a theoretical analysis that assumes a completely uniform communication pattern. Figure 6 also gives a clear view of how those cores that suffer longer latencies in the mesh are the most benefited by

DOI: 10.1002/cpe

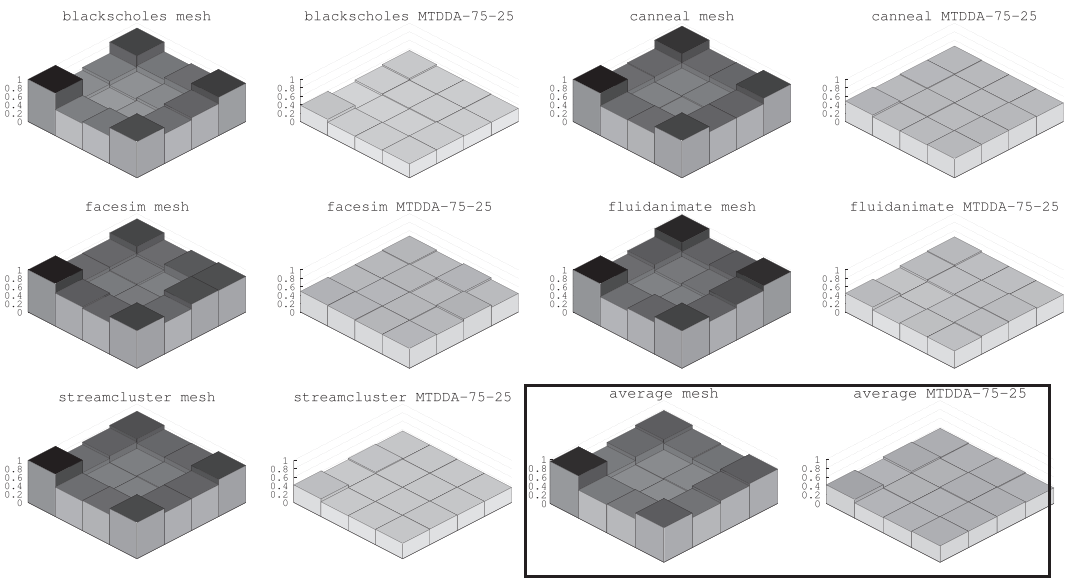
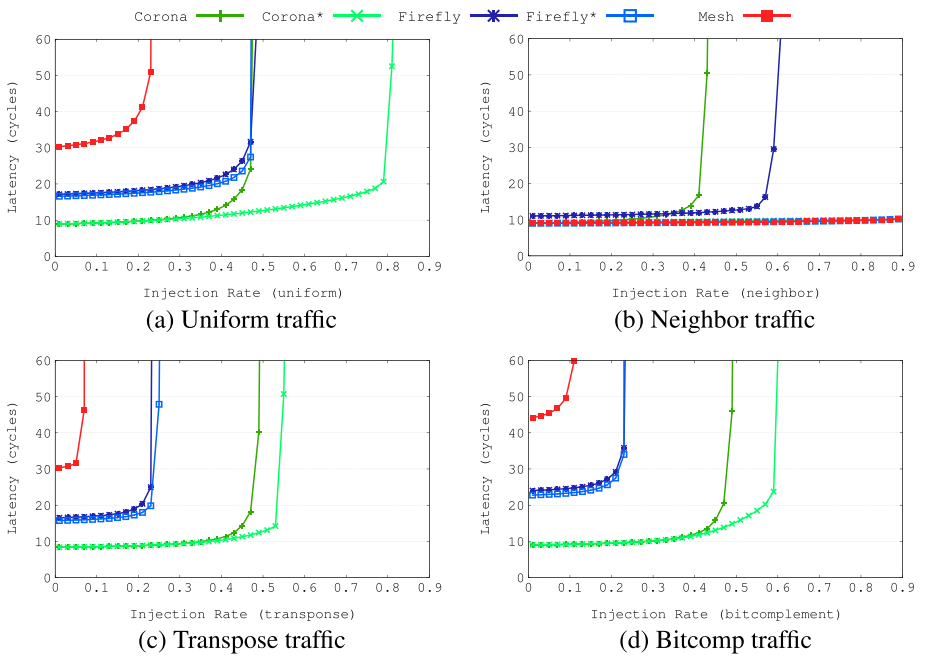


Figure 6. Total network latency suffered by each core in the critical path of L1 cache misses. Results for electrical mesh and MTDDA-75-25 for selected PARSEC benchmarks and average of all benchmarks. The data are normalized to the core with the highest network latency in the mesh.



(a) Uniform traffic (b) Neighbor traffic

(c) Transpose traffic (d) Bitcomp traffic

Figure 7. Load latency for uniform (a), neighbor (b), transpose (c), and bitcomp traffic (d). For a 256-bit photonic datapath.

MTDDA-75-25 (corners and borders of the chip). The threads running on these cores obtain a higher speed up, and their performance can now match that of the threads running in the central cores. Figure 3 also shows that our policies reduce the standard deviation of the network latency suffered by cores. This reduction is especially noticeable for DDA and MTDDA where it reaches 60%. This means that a much more homogeneous network latency is perceived by all cores, which has the important benefit of reducing the wait times for thread synchronization. This provides a noticeable portion of the acceleration of applications seen in Figure 3 by reducing wait times between threads.

DOI: 10.1002/cpe

4.4. Results for 64-endpoint hybrid NoCs

For these large networks, we have chosen to show only the results of the MTDDA-75-25 policy managing Firefly and Corona. We use the names *Firefly** and *Corona** to refer to the hybrid networks managed by MTDDA-75-25, in contrast to *Corona* and *Firefly*, which we use to refer to the original NoCs. We also simulated one of the simpler policies, AVAIL-6, as a baseline to compare against MTDDA-75-25 and determine which benefits come from the extra network resources and which from a smarter policy. We do not show AVAIL-6 in the graphs for clarity, but we refer to the hybrid networks managed by AVAIL-6 by the names *CoronaAV* and *FireflyAV* in our analysis.

Figure 7 shows the results for four synthetic traffic patterns. Under uniform traffic, Corona* throughput (0.80 msgs/cycle) is 18% larger than the sum of Corona (0.46 msgs/cycle) and the mesh (0.22 msgs/cycle) separately. MTDDA-75-25 exploits the best features of the mesh for short distance transmission, using it for close destinations (fast transmission and high throughput) and avoiding distant ones (slow transmission and low throughput), for which MTDDA-75-25 uses photonics (fast and same throughput regardless of distance). This increases the overall throughput of the hybrid network, and at the same time keeps a very low latency for Corona*. CoronaAV is not so efficient because it does not take distance into account, resulting in a throughput (0.58 msgs/cycle) higher than Corona but lower than the sum of Corona and the additional electronic resources, and higher latency generally close to the mesh.

Neighbor traffic benefits from the lowest latency and highest throughput possible in a mesh (>1.00 msgs/cycle). Corona (0.42 msgs/cycle) and Firefly (0.60 msgs/cycle) yield poor throughputs (Firefly has an advantage against Corona because of the intra-cluster links) compared to Corona* and Firefly* (>1.00 msgs/cycle), which benefit from the electronic links to neighbours. The 32 extra links of Firefly* make a big difference over Firefly. CoronaAV and FireflyAV also have high throughput, but their latency is 50–60% higher than Corona* and Firefly* due to the photonic waiting times introduced by AVAIL-6.

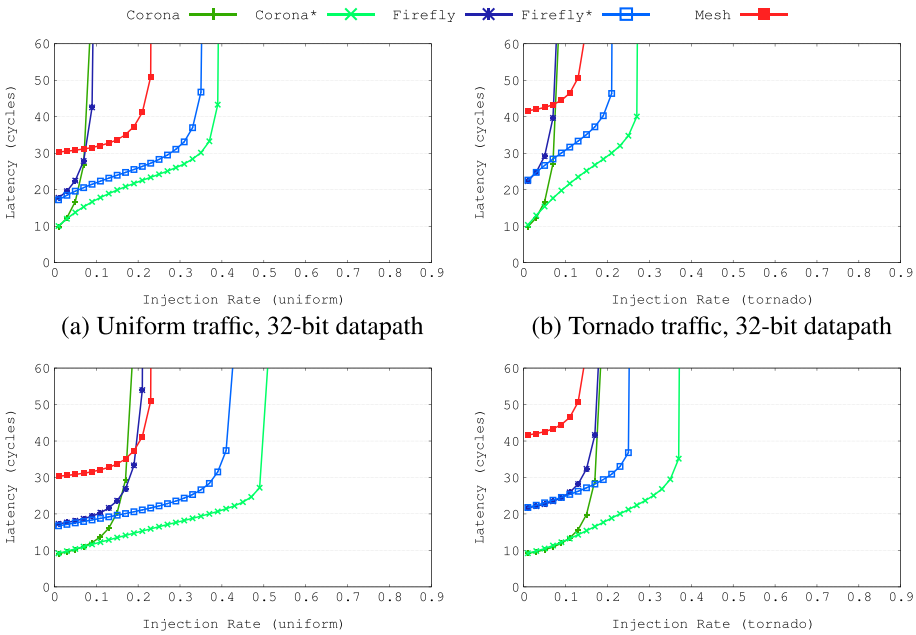
In transpose and bitcomp traffics, Corona* (0.54, 0.60 msgs/cycle) does not achieve great improvements because each origin has a fixed destination, giving less flexibility for MTDDA-75-25 to arbitrate. Yet Corona* provides some advantage over CoronaAV (0.50, 0.51 msgs/cycle) and Corona (0.48, 0.48 msgs/cycle), because MTDDA-75-25 uses the mesh efficiently for those origin-destination pairs with shorter distances and, in any case, because it takes into account the dynamic photonic-link occupation status to decide the most-effective route for each message.

Firefly* and FireflyAV do not benefit from the extra 32 links over Firefly in uniform, transpose, bitcomp nor tornado traffics, confirming the observations by Firefly's authors. The Firefly design forces electronic intra-cluster transmission for every message (except when a direct photonic path exists, which happen only in seven of 63 cases), flooding the intra-cluster electronic links. This happens also to Firefly* and FireflyAV. In fact, this intra-cluster bottleneck limits the achievable degree of utilization of the inter-cluster photonic resources, which never rises over 37% in transpose traffic.

For completeness in our analysis, we have also simulated versions of the Corona and Firefly networks with a photonic datapath width reduced from 256 bits to 64 and 32 bits. These configurations are intended to fill the gap in our analysis between the short-term ring evaluated in Section 4.1 and the long term proposals of Firefly and Corona just evaluated. Figure 8 shows the results for uniform and tornado traffic for 32-bit and 64-bit datapaths. The electric mesh shows superior or similar throughput to Corona and Firefly for these photonic datapath sizes.

In uniform and tornado traffics with a 32-bit datapath, Corona* (0.39, 0.28 msgs/cycle) has 30% and 33% higher throughput than the sum of Corona (0.08, 0.07 msgs/cycle) and the mesh (0.22, 0.14 msgs/cycle). More importantly, MTDDA-75-25 uses photonics smartly (for long distances) managing to keep the latency of the hybrid network much lower than that of the mesh, which is providing most of the throughput (for short distances, with low latency). CoronaAV (with AVAIL-6) does not do this, resulting in average latencies higher than the mesh and lower throughputs (0.25, 0.17 msgs/cycle) than Corona*. We can conclude that MTDDA-75-25 increases the throughput of the hybrid network noticeably, over the sum of the parts, and achieves efficient latencies with any combination of photonic and electronic resources, while less elaborated policies

DOI: 10.1002/cpe



(a) Uniform traffic, 32-bit datapath (b) Tornado traffic, 32-bit datapath

(c) Uniform traffic, 64-bit datapath (d) Tornado traffic, 64-bit datapath

Figure 8. Load latency for uniform and tornado traffic, using 32-bit (top) and 64-bit (bottom) photonic datapath.

Table VII. Percentage of packets optically transmitted for each distance (network hops). Uniform traffic, 0.15 msgs/cycle injection rate.

	1	2	3	4	5	6	7	8	9	10	11	12	13	14
Firefly	29	59	81	94	100	100	100	100	100	100	100	100	100	100
Firefly*	0	13	28	49	64	70	75	79	82	84	85	85	86	88

cannot do any of this. Another representative result is that under uniform traffic, Corona* with a 64-bit datapath (0.49 msgs/cycle) has slightly higher throughput than Corona with a 256-bit datapath (0.48 msgs/cycle), while CoronaAV with the same 64-bit datapath performs far worse (0.35 msgs/cycle).

Note that Firefly contains most of the mesh, but its throughput with a 32-bit photonic datapath is noticeably worse than the mesh. The design of Firefly forces the use of photonics for most messages, even between neighbours, making photonics the limiting factor, and misusing the intra-cluster electronic links. This is the opposite case to the one noted with a larger 256-bit photonic datapath, in which the intra-cluster links were the bottleneck limiting the efficient use of photonics. Table VII shows the percentage of messages that used photonics in uniform traffic at an injection rate of 0.15 msgs/cycle. The absence of inter-cluster links makes Firefly use photonics for already 29% of one-hop transmissions and for most two-hop transmissions, creating the bottleneck. In contrast, Firefly* does not need photonics for one-hop transmissions, using the inter-cluster links instead, and the usage of photonics grows gracefully with the distance. With uniform traffic and a 32-bit datapath, Firefly*'s (0.33 msgs/cycle) throughput is four times larger than Firefly (0.08 msgs/cycle) , by just adding the 32 extra inter-cluster links.

5. RELATED WORK

Since silicon-photonic integration became a feasible solution for CMP interconnects, a myriad of photonic networks have been proposed as a solution to the lack of scalability of electrical NoCs.

DOI: 10.1002/cpe

Many nanophotonic-based network topologies have been studied, from simple photonic rings [6–8] that operate like a crossbar to complex articulated topologies [9, 10] that require or combine different transmission technologies, and to logical all-to-all interconnect designs [11]. Instead of proposing new topologies, our work is the first to provide fine-grain policies to exploit the best features of nanophotonics and electronics working together.

Some complex photonic interconnects use supporting circuit-establishing electrical networks [9, 10]. This limits severely the latency and energy advantages of nanophotonics in scenarios like hardware-cache-coherent CMPs with memory-block-grain network communication. We focus on a simple photonic structure (ring) instead and show that, if properly managed, it can potentially deliver large latency and energy improvements without needing big investments in complex topologies and/or organization.

Some sort of hybrid NoC design is often assumed. Concentration is present in many works [6–8, 30]. That means that some processing elements share each photonic endpoint, using electronic communication between them. These interconnects use both electric and photonic technologies, whose interaction is decided at design time. Like them, we focus on realistic traffic-constricted photonic baselines, but unlike them, we focus on a more efficient dynamic management of the electric/photonic NoC resources at a message granularity.

Also, different arbitration and QoS policies [31, 32] have been proposed, as well as static traffic selection policies [33], to make the most of the limited resources of the shared photonic medium. On our part, we have shown how dynamic management policies based on distance between endpoints enable more fairness in the chip by reducing the network latency differences suffered by tiles in different positions of the chip.

6. CONCLUSIONS

In this paper, we have shown the importance of using adequate management policies to enable efficient use of any amount of photonic resources in an hybrid photonic-electronic network. We have proposed the first dynamic fine-grain policies to enable such management. We have tested these policies both on a near-future affordable photonic ring and on far larger ring-based photonic networks, obtaining significant performance improvements and energy consumption reductions.

The proposed message-granularity policies are based on distance between endpoints, ring availability and message size. By using photonics for the messages most likely to benefit from it (distant, short, and keeping low waiting times), we reduce the number of electric mesh retransmissions (that cause large energy consumption and latency). At the same time, we prevent severe message serialization on the photonic ring by resorting to the mesh when necessary, and preferably for short-distance messages. In addition, these policies level out the network latency suffered by all cores in the chip compared to an electrical mesh, resulting in an additional performance boost thanks to quicker synchronization and lower processor idle time. A proposed performance-oriented policy (CDDA-75) reduces execution time by 36%, and a proposed energy oriented policy (DDA-75) reduces network energy consumption by 52% for the PARSEC benchmark suite in a 16-core CMP. In addition, we propose a balanced policy (MTDDA-75-25), which reduces execution time by 35% and reduces network energy consumption by 48%. Larger NoCs also show far superior throughput and lower latency if managed by an appropriate policy.

ACKNOWLEDGEMENT

Part of this work was performed by Antonio García Guirado during a research stay at the University of Siena (February–August 2012) partially supported by a collaboration grant of the HiPEAC NoE (IST-217068). This work was also supported by the Spanish MINECO and Spanish MEC, as well as European Commission FEDER funds under grant number TIN2012-31345, and by IT FIRB Photonica (RBFR08LE6V). Antonio García-Guirado was also supported by a research grant from the Spanish MEC under the FPU National Plan (AP2008-04387).

DOI: 10.1002/cpe

REFERENCES

1. Kim C, Burger D, Keckler SW. An adaptive, non-uniform cache structure for wire-delay dominated on-chip caches. *Proceedings of the 10th International Conference on Architectural Support for Programming Languages and Operating Systems (ASPLOS)*, San Jose, CA, USA, 2002; 211–222.
2. Magen N, Kolodny A, Weiser U, Shamir N. Interconnect-power dissipation in a microprocessor. *Int'l Workshop on System Level Interconnect Prediction (SLIP)*, Paris, France, February 2004; 7–13.
3. Borkar S, Chien AA. The future of microprocessors. *Communications of the ACM* May 2011; **54**(5):67–77.
4. Gunn C. CMOS photonics for high-speed interconnects. *IEEE Micro* March 2006; **26**(2):58–66.
5. Jalali B, Fathpour S. Silicon photonics. *Journal of Lightwave Technology* December 2006; **24**(12):4600–4615.
6. Vantrease D, Schreiber R, Monchiero M, McLaren M, Jouppi NP, Fiorentino M, Davis A, Binkert N, Beausoleil RG, Ahn JH. Corona: system implications of emerging nanophotonic technology. *Proceedings of the 35th Annual International Symposium on Computer Architecture (ISCA)*, Beijing, China, 2008; 153–164.
7. Pan Y, Kim J, Memik G. Flexishare: channel sharing for an energy-efficient nanophotonic crossbar. *Proceedings of the 16th International Symposium on High-Performance Computer Architecture (HPCA)*, IEEE CS, Bangalore, India, 2010; 1–12.
8. Xu Y, Du Y, Zhang Y, Yang J. A composite and scalable cache coherence protocol for large scale CMPs. *Proceedings of the International Conference on Supercomputing (ICS)*, Tucson, AZ, USA, 2011; 285–294.
9. Shacham A, Bergman K, Carloni LP. Photonic networks-on-chip for future generations of chip multiprocessors. *IEEE Transactions on Computers* September 2008; **57**(9):1246–1260.
10. Petracca M, Lee BG, Bergman K, Carloni LP. Design exploration of optical interconnection networks for chip multiprocessors. *16th IEEE Symposium on High Performance Interconnects*, Stanford, CA, USA, 2008; 31–40.
11. Nitta C, Farrens M, Akella V. DCAF - a directly connected arbitration-free photonic crossbar for energy-efficient high performance computing. *26th International Parallel & Distributed Processing Symposium (IPDPS)*, Shanghai, China, 2012; 1–12.
12. Pan Y, Kumar P, Kim J, Memik G, Zhang Y, Choudhary A. Firefly: illuminating future network-on-chip with nanophotonics. *Proceedings of the International Symposium on Computer Architecture (ISCA)*, Austin, TX, USA, 2009; 429–440.
13. Hendry G, Kamil S, Biberman A, Chan J, Lee BG, Mohiyuddin M, Jain A, Bergman K, Carloni LP, Kubiatowicz J, Oliker L, Shalf J. Analysis of photonic networks for a chip multiprocessor using scientific applications. *Proceedings of the 2009 3rd ACM/IEEE International Symposium on Networks-on-Chip (NOCS)*, IEEE Computer Society, Washington, DC, USA, 2009; 104–113.
14. Li Z, Fay D, Mickelson AR, Shang L, Vachharajani M, Filipovic D, Park W, Sun Y. Spectrum: a hybrid nanophotonic-electric on-chip network. *46th Annual Design Automation Conference*, San Francisco, CA, USA, 2009; 575–580.
15. Bahirat S, Pasricha S. A particle swarm optimization approach for synthesizing application-specific hybrid photonic networks-on-chip. *13th International Symposium on Quality Electronic Design*, IEEE, Santa Clara, CA, USA, 2012; 78–83.
16. Bogaerts W, De Heyn P, Van Vaerenbergh T, De Vos K, Kumar Selvaraja S, Claes T, Dumon P, Bienstman P, Van Thourhout D, Baets R. Silicon microring resonators. *Laser & Photonics Reviews* 2012; **6**(1):47–73.
17. Xu Q, Fattal D, Beausoleil RG. Silicon microring resonators with 1.5-μm radius. *Optics Express* 2008; **16**: 4309–4315.
18. Zhang L, Yang M, Jiang Y, Regentova E, Lu E. Generalized wavelength routed optical micronetwork in network-on-chip. *Proceedings of 18th IASTED International Conference on Parallel and Distributed Computing and Systems*, Dallas, TX, USA, 2006; 698–703.
19. O'Connor I, Van Thourhout D, Scandurra A. Wavelength division multiplexed photonic layer on cmos. *Proceedings of the 2012 Interconnection Network Architecture: On-Chip, Multi-Chip Workshop*, Paris, France, 2012; 33–36.
20. Martin MMK, Hill MD, Sorin DJ. Why on-chip cache coherence is here to stay. *Communications of the ACM* 2012; **55**:78–89.
21. Batten C, Joshi A, Stojanovic V, Asanovic K. Designing chip-level nanophotonic interconnection networks. *IEEE Journal on Emerging and Selected Topics in Circuits and Systems* 2012; **2**(2):137–153.
22. Kirman N, Kirman M, Dokania RK, Martinez JF, Apsel AB, Watkins MA, Albonesi DH. Leveraging optical technology in future bus-based chip multiprocessors. *Proceedings of the 39th IEEE/ACM International Symposium on Microarchitecture (MICRO)*, Orlando, FL, USA, 2006; 492–503.
23. Xiao C, Frank Chang M-C, Cong J, Gill M, Huang Z, Liu C, Reinman G, Wu H. Stream arbitration: towards efficient bandwidth utilization for emerging on-chip interconnects. *ACM Transactions on Architecture and Code Optimization* January 2013; **9**(4):article 60, 1–27.
24. García-Guirado A, Fernández-Pascual R, Ros A, García JM. DAPSCO: distance-aware partially shared cache organization. *ACM Transactions on Architecture and Code Optimization* January 2012; **8**(4):25:1–25:19.
25. Binkert N, Beckmann B, Black G, Reinhardt SK, Saidi A, Basu A, Hestness J, Hower DR, Krishna T, Sardashti S, Sen R, Sewell K, Shoaib M, Vaish N, Hill MD, Wood DA. The gem5 simulator. *SIGARCH Computer Architecture News* August 2011; **39**(2):1–7.
26. Kahng AB, Li B, Peh L-S, Samadi K. ORION 2.0: a fast and accurate NoC power and area model for early-stage design space exploration. *Proceedings of the Conference on Design, Automation and Test in Europe (DATE)*, Nice, France, 2009; 423–428.

DOI: 10.1002/cpe

27. Zheng X, Patil D, Lexau J, Liu F, Li G, Thacker H, Luo Y, Shubin I, Li J, Yao J, Dong P, Feng D, Asghari M, Pinguet T, Mekis A, Amberg P, Dayringer M, Gainsley J, Moghadam HF, Alon E, Raj K, Ho R, Cunningham JE, Krishnamoorthy AV. Ultra-efficient 10gb/s hybrid integrated silicon photonic transmitter and receiver. *Optics Express* 2011; **19**(6):5172–5186.
28. Grani P, Bartolini S. Design options for optical ring interconnect in future client devices. *ACM Journal on Emerging Technologies in Computing Systems* 2014; **10**(4):article 30, 1–25.
29. Fallin C, Nazario G, Yu X, Chang K, Ausavarungnirun R, Mutlu O. MinBD: minimally-buffered deflection routing for energy-efficient interconnect. *Proceedings of the 2012 IEEE/ACM Sixth International Symposium on Networks-On-Chip (NOCS)*, NOCS '12, Lyngby, Denmark, 2012; 1–10.
30. Kurian G, Miller JE, Psota J, Eastep J, Liu J, Michel J, Kimerling LC, Agarwal A. ATAC: a 1000-core cache-coherent processor with on-chip optical network. *Proceedings of the 19th International Conference on Parallel Architectures and Compilation Techniques (PACT)*, Vienna, Austria, 2010; 477–488.
31. Vantrease D, Binkert N, Schreiber R, Lipasti MH. Light speed arbitration and flow control for nanophotonic interconnects. *Proceedings of the 42th Annual IEEE/ACM International Symposium on Microarchitecture (MICRO)*, New York, NY, USA, 2009; 304–315.
32. Pan Y, Kim J, Memik G. FeatherWeight: low-cost optical arbitration with QoS support. *Proceedings of the 44th Annual IEEE/ACM International Symposium on Microarchitecture (MICRO)*, Porto Alegre, Brazil, 2011; 105–116.
33. Bartolini S, Grani P. A simple on-chip optical interconnection for improving performance of coherency traffic in CMPs. *15th Euromicro Conference on Digital System Design (DSD)*, Cesme, Turkey, 2012; 312–318.

DOI: 10.1002/cpe