A Simple On-chip Optical Interconnection for Improving Performance of Coherency Traffic in CMPs

Sandro Bartolini, Paolo Grani
Dipartimento di Ingegneria dell’Informazione
Università degli Studi di Siena
via Roma 56, 53100, Siena, Italia
{bartolini,grani}@di.unisi.it

Abstract—Nanophotonic interconnection is a promising solution for inter-core communication in future chip multiprocessors (CMPs). Main benefits derive from its intrinsic low-latency and high-bandwidth, especially when employing wavelength division multiplexing (WDM), as well as reduced power requirements when compared to electronic NoCs. Existing works on optical NoCs (ONoC) mainly concentrate on relatively complex proposals needed to host the whole CMP traffic. In some proposals complexity is increased also from the need of an electronic network for preliminary path-setup in the optical one. This paper proposes to enhance a conventional NoC with only a simple photonic structure, a ring, and aims at investigating its suitability to support the low-latency transmission of small latency-critical coherency control messages as to improve performance of multithreaded applications. In particular, our proposed scheme supports fast multicast transmission of invalidation messages. We have simulated Parsec benchmarks on an 8 core full-system CMP. Results show that a careful selection of coherency control messages to be forwarded to the photonic ring allows improving execution time up to 19%, with an average of 6% across all considered benchmarks. We discuss how different selections of messages, i.e. related to read and/or write operations, affect results and single out the most profitable set. Moreover, we show that the sharing behavior of benchmarks has a central role in the final performance.

Keywords—Chip Multiprocessors, Coherency, Multithreaded Applications, On-chip Photonics, Selective traffic management.

I. INTRODUCTION

State-of-the-art commercial processors already comprise numerous cores into a single die (e.g. up to 16 in recent AMD Opteron 6200 models [1]). Prototype processors like the Intel Polaris [2] reached 1 TFlop some years ago thanks to an 80 core tiled architecture built around an ad-hoc interconnection network. Tiled CMP architectures are considered a promising scalable design paradigm [3] as they replicate identical building blocks (tiles) and connect them through a Network on Chip (NoC). AMD Bulldozer family [1] is a nowadays commercial example of the tiled approach. In order to translate the potential computational power of a set of tiles into parallel application performance, an efficient on-chip cache hierarchy [4] and communication infrastructure has to be provided. For this reason research on efficient NoC designs is absolutely important. Current and future CMPs running modern multi-threaded applications require high-bandwidth and low-latency interconnection for efficiently managing coherency protocol evolution. Also due to the emerging wire delay issues [5] traditional electronic NoC (eNoC) designs have troubles in fulfilling these requirements while maintaining acceptable power consumption [6]. On-chip nanophotonic technologies are now considered a viable solution for fulfilling future system bandwidth demands (for both intra-chip and inter-chip data transfers), low latency communication and reduced power consumption [7]. Low latency trait originates from both the fast transmission speed of light into silicon\(^1\) and from the ability to transmit even across the whole chip without the need of regenerating the signal as in eNoCs. Existing works have analyzed various on-chip photonic networks [8]–[11] as an alternative interconnection technology to traditional eNoCs and, for this reason, such networks architectures need to be quite complex in order to sustain all the CMP traffic. In this paper we evaluate the potential benefits originating from a simple on-chip photonic structure, a ring, used in parallel (i.e. 3D stacked) with a standard electronic NoC. In this way, we will be able to assess how a limited investment in on-chip photonic technology can improve CMP performance and, consequently, this study will allow to add an additional perspective in the understanding of the correlation between the various existing ONoC approaches and resulting performance results. As the ring structure cannot efficiently serve the overall traffic, the idea is to provide a low-latency and high-bandwidth path only to the latency critical control communications required to maintain cache coherency. Specifically, we will measure the performance improvement margin of multi-threaded applications when specific subsets of coherency messages are routed onto the photonic path, while all the remaining traffic (control + data) is kept on the electronic network. Recent studies show that even though latency-critical traffic is only a small portion of the total network traffic, if it is not properly handled, system performance can degrade significantly [12].

\(^1\) About 66600 km/s → 15 ps/mm
is organized as follows. In section II we briefly introduce the technologies addressed by the paper. In section III, we describe the features of the considered architecture, while in section IV we present the evaluation methodology and discuss the achieved results. In section V we present some related works and in section VI we point out some future work. Finally, section VII concludes.

II. BACKGROUND

In this section we recall some background on coherency protocols for CMPs and on-chip photonic interconnection. Fig. 1 gives a logical overview of our baseline CMP architecture where each core has private L1 caches (Instruction+Data) and a shared L2 cache, distributed in the chip so that each core has a slice of L2 cache close to it. Directory information is distributed as well and is stored along each L2 cache slice. A traditional mesh eNoC allows the communication between L2 slices and, consequently, between cores. Fig. 1 highlights the photonic ring path, super impressed in grey, which is integrated into the baseline architecture to obtain our enhanced one.

A. Coherency protocols over a NoC

In this work we rely on MOESI coherency protocol, which is a full cache coherency protocol that encompasses all of the states of the well-known MESI protocol, plus a fifth "Owned" state representing data that is both modified and shared. This state avoids the need to write a dirty cache line back to the next memory level when another processor tries to read it. In fact the "Owner" node, that holds the valid copy of a block, can supply the data directly to the requesting core. The specific solutions proposed in this work are quite general and, for example, can fit also the MESI protocol. In a CMP, coherency protocol evolution over a NoC can require numerous message transmissions according to the state of the involved modules (i.e. L1 and L2 caches, and directories). Many of such messages are short control messages and don’t carry data. Fig. 2 gives two examples of coherency message sequences as consequences of two actions done at L1 level from a processor. Fig. 2a shows a L1 read miss to a shared data in the leftmost L1 cache. The miss induces a request for shared data (GETS) to the "home" L2+directory slice, i.e. the middle one in fig. 2a, which doesn’t have the copy and, in turn, asks the owning L1 to forward the data to the requestor. Fig. 2b shows a write miss to a shared data. The GETX request causes the home L2+directory slice (leftmost slice) to send invalidations towards the sharers and, among them, stimulates the rightmost L1, which is in owned state, to forward the data to the requesting core. From the examples above it turns out that basic core operations (Load, Store) can induce quite articulated sequences of protocol operations and NoC messages, involving also conceptually multicast messages (invalidations). From another perspective, the time to handle such protocol operations and message delivery

2 L2 bank and directory portion that are responsible of caching the data in L2 and managing MOESI state, respectively.
determines the execution time of processor operations and are perceived by the processor as part of the memory hierarchy latency. Thus, speeding up the protocol evolution through a faster delivery of coherency control messages can positively impact the overall CMP performance.

B. On-chip optical interconnection

On-chip optical interconnection technology is based on the effective possibility to integrate waveguides, lasers, modulators and photo-detectors into current process technology [7]. In line with existing proposals for complex on-chip optical networks [9], [13], we assume to host our ring optical path on a dedicated layer of a 3D stacked chip architecture. The communication paradigm, diversely from electronic networks, is point-to-point by nature, as light packets cannot be managed in a store-and-forward fashion within the optical domain. Therefore, photonic links can be effectively used to directly connect communication endpoints. This feature determines one of the main benefits of this technology: end-to-end communication latency is extremely low because it is dominated by light propagation speed into silicon. Moreover, power consumption of a transmission is, in practice, independent from the on-chip distance. The achievable modulation bandwidth can easily reach 10 GHz. Passive and active optical switch architectures have been proposed for implementing network topologies with fixed (passive) [10], [14] and reconfigurable (active) [7] end-to-end paths. Passive network tend to require more optical modules than active ones in order to provide similar connectivity. However, active ones require preliminary optical path setup, which has to be performed through a parallel electronic network and can incur in non-negligible setup time due to micro-ring thermal tuning.

III. Proposed Architecture

Our hybrid electro-optical NoC architecture is a standard electronic mesh augmented with a simple photonic ring (3D stacked) which connects all L2+directory modules and implements a low-latency and high-bandwidth path between them. The ring is supposed to help in supporting the transmission of coherency messages efficiently. Photonic transmission can natively implement multicast communication through WDM, in which multiple wavelengths, associated to different destinations, can be contemporarily routed onto the same waveguide without interfering with each other. This can be done exploiting the same waveguide, using multiple wavelengths at the same time, depending on how many destinations have to be invalidated. We aim at exploiting the possibility to map invalidation messages, which are conceptually multicast to reach all the sharers [12], onto this technological opportunity. Moreover invalidation in a full system simulation can experience up to twice the latency than the average of all messages [12], so it is crucial to improve their transmission time. The eNoC is connected to the photonic ring through an additional port in each electronic switch and by Through-Silicon Vias (TSVs) towards the 3D stacked photonic layer. The main benefit of a 3D stacked design is the ability to easily integrate heterogeneous technologies in the same chip. In our reference configuration we use WDM to implement a conceptually passive routing, in which we associate a wavelength to every optical destination. Whenever a L2+directory node has to send a message to another node, or to a set of nodes (invalidations), it modularizes the associated wavelength or wavelengths, respectively. This scheme is quite scalable as WDM can currently reach 64 wavelengths [10] (MaxWDM=64), thus accommodating up to 64 cores. Furthermore, if cores are clustered into small groups (e.g. 4), connected via local electronic links, and one hub per cluster is provided to reach the photonic layer, further scalability is potentially achievable. In this work we aim at evaluating the proposed idea in a non-clustered architecture in order to highlight its intrinsic pros and cons. We leave to future work the analysis of clustered approaches. Actually, in this scheme each sending node can use up to \( \lfloor MaxWDM/n \rfloor \) wavelengths to send more than one bit at a time into the waveguide, where \( n \) is the total number of nodes in the system. Therefore, as we evaluate an 8-node architecture and MaxWDM=64, we dedicate 8 wavelengths to each destination so that optical communications can physically happen with 8-bit parallelism. The routing mechanism adopted in this work to permit the correct switching between the narrow photonic ring (8 bit per core) and the wide electronic mesh (128 bit per core) is a specifically tuned wormhole technique [15]. In our system, messages carrying data are 72-byte long, control messages are 8-byte long and flit size in the eNoC is 16-byte. Our objective is to exploit the photonic ring to quickly route to
destination a set of coherency control messages in order to reduce the latency that they induce in the execution of multi-threaded applications. In this setup we found that if all the messages (control and data), or even only data messages, are forwarded onto the photonic ring, execution time slows down significantly, due to the overexploitation of the ring itself. Fig. 3 shows the average execution time of all the Parsec benchmarks for the baseline eNoC configuration and for the hybrid eNoC+photonic ring architecture with different bit (i.e. wavelength) parallelism (1 bit, 2 bit, 4 bit and 8 bit), when all the traffic is routed onto the photonic path (light grey) as well as when only the data messages are transmitted on the photonic ring (dark grey). More bits (wavelengths) per destination allow for higher bandwidth and performance but even with 8-bit per destination the traffic saturates the photonic channel and induces a net 2x slowdown. The dark grey bars witness that even data messages alone cause the overexploitation of the photonic ring and this, in turn, prevents from taking advantage of the potential intrinsic low-latency optical communication. For this reason, it is crucial to route on the photonic ring only small and latency-critical control messages in order to take the most out of the simple ring photonic structure and reduce the overall coherency latency. In case of write requests, invalidation process can require sending messages (Inv messages) to multiple destinations and wait acknowledgement (Inv-Ack messages) from them. This can induce a significant delay in solving write requests and test-and-set operations. In sending invalidations, we also rely on the multicast capabilities of the photonic technology through WDM. We have selected three promising groups of control messages for photonic routing, all comprising Inv and Inv-Ack messages. The first configuration routes also write requests from the initiator of the write (GETX message). The second configuration routes read requests (GETS message) but not write requests, thus accelerating both reads through GETS management and writes through Inv and Inv-Ack messages. The last configuration routes on the photonic path both GETS and GETX messages.

IV. Results

In this section we introduce our evaluation methodology and test setup, and discuss the achieved results.

A. Methodology and test setup

Performance evaluation are obtained using the gem5 simulator [16], in which we model a full system 8 core CMP architecture with private L1 instruction and data caches and a distributed shared L2 cache. In the baseline architecture, L2 sub-banks are interconnected through a 2D mesh network, while the enhanced architecture has an additional 3D stacked photonic ring. The photonic ring has 8 I/O ports, each connecting to a standard electronic switch and so to a L2+Directory module. In this way, some messages can be selectively routed to the photonic path for reduced latency and efficient broadcasting. Table I summarizes some details of the considered architecture. Cache parameters are derived by CACTI 6 [17]. Parameters of the photonic devices are in line with state-of-the-art proposals. For a commercially standard chip size, the end-to-end optical communication over the ring can be estimated within 460 ps (@15ps/mm propagation speed), comprising modulator+driver, waveguide, photodetector and receiver [9]. According to this, a typical coherency control message can be transmitted over the full ring length, i.e. worst-case, in about 6.8ns (less than 28 cycles@4GHz) using one wavelength. Using 8 wavelengths per destination, such transmission time becomes 1.3ns (5 cycles@4GHz). We also rely on the possibility to multicast invalidations using WDM. Performance was gathered for the Parsec 2.1 benchmark suite [18], [19], a collection of heterogeneous multi-threaded applications spanning different application domains. Applications rely on Linux 2.6.27 operating system (OS), which is booted onto the simulated architecture. All benchmarks were instantiated for a degree of parallelism of 8 threads, one per core. Benchmarks were modified to enforce core affinity as to avoid non determinism due to OS scheduling. We use the small input set, which however translates into a pretty significant number of executed instructions: from 0.85 billion of blackscholes up to 24.9 billion of facesim that, actually, has only one input set, the large one. As for the performance metrics, we considered the execution time of the entire parallel region of each benchmark as representative of the end-user perceived performance. Then, the overall L2+network latency was used to get insight on how the proposed hybrid electronic-optical network is able to improve the average performance of the on-chip communication and caching infrastructure for the operations that cannot be solved in the private L1 caches.

B. Result discussion

Before analyzing execution time results we discuss the latencies experienced by messages on the network. Fig. 4a compares the NoC behavior between baseline architecture and extended one when both GETS and GETX messages are routed onto the photonic path. The average network latency improvement is of about 10% for both the overall

<table>
<thead>
<tr>
<th>Table I</th>
<th>PARAMETERS OF THE SIMULATED ARCHITECTURE</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cores</td>
<td>8 Alpha processors, 4 GHz</td>
</tr>
<tr>
<td>L1 caches</td>
<td>16 kB (I) + 16 kB (D), 2-way, 1 cycle hit time</td>
</tr>
<tr>
<td>L2 cache</td>
<td>16 MB, 8-way, shared and distributed 8 2MB banks, 3/12 cycles tag/tag+data</td>
</tr>
<tr>
<td>Directory</td>
<td>MORF protocol, 8 slices, 3 cycles</td>
</tr>
<tr>
<td>NoC</td>
<td>2D-Mesh, 4 GHz, 4 cycles/hop</td>
</tr>
<tr>
<td>Photonic ring</td>
<td>3D-stacked, 30mm length, 8 I/O ports, 10 GHz, 64 WDM, 640ps full round path</td>
</tr>
<tr>
<td>Main memory</td>
<td>4 GB, 300 cycles</td>
</tr>
</tbody>
</table>
and reads (LD) messages, respectively, and of about 7% for messages related to writes (ST). As it will be clear in the analysis later in this section, the average latency of these network operations does not exactly translate into an equal application speedup since other factors can degrade the execution time improvement. Most of them relate to interactions between core operations and coherency protocol, possible race conditions, interactions between synchronization code and the operating system itself. This originates from the complexity of the full-system CMP simulation of realistic multithreaded benchmarks, which is needed to model and evaluate in details current and future multiprocessor architectures. Based on MOESI protocol behavior (see Fig. 2), we have aimed at reducing the critical path of the write requests that require invalidation of sharers, which induce a long sequence of message exchanges. So, we have routed onto the photonic link invalidation messages (Inv), the corresponding acknowledgements (Inv-Ack) and the GETX message from the requesting L1. In brief, most of the control messages during a write operation flow very fast and the possibly multiple invalidations can be overlapped in time (multicast). Results of this setup are shown in the second bars of Fig. 4b (GETX bar) which show that vips and dedup reach more than 13% improvement, blacksholes speeds up by 11% and x264 improves by almost 8%. Apart from fluidanimate, which show a 5% improvement, the remaining benchmarks show a limited speedup (within 3%) and two, canneal and streamcluster experience a slight slow-down, within 3%, due to the specific nature of their access pattern, especially to shared data. In particular, canneal is known to have a strong and fine-grained all-to-all random communication pattern and has a big shared working set [19]. These features induce a, quite singular, dense synchronization activity due to operations for mutual exclusion acquisition [20]. In particular, we have singled out that the few benchmarks that don’t improve much or incur in slight slowdown when using the photonic enhanced network (e.g. canneal), suffer more than the others from the increase of race conditions in the home L2-slice due to spinlocks. In fact, thanks to the overall low-latency and to the relatively uniform latency of the photonic path from every node to reach any other one (e.g. the home node), multiple invalidated nodes that want immediately reload the data and try to acquire a lock, have higher probability to interleave and make other’s test-and-set operation fail. This is not an issue in itself but we found out that it can translate into a significant time overhead because of the interaction between the synchronization mechanisms in the pthread+semaphore libraries and the operating system, which can decide to de-schedule a thread that fails a number of test-and-set in a row. Indeed, some of these issues are mainly related to the shared-memory environment, which is however dominant in non-application-specific systems. The study of message passing environments is out of the scope of this work. When we route onto the photonic ring read-related control messages (GETS), invalidations (Inv) and invalidation acknowledgements (Inv-Ack), as shown in the third bars of Fig. 4b (GETS bar), overall results are slightly better than in the previous case, with an average improvement of 6% over the electronic NoC. In this configuration the selective photonic routing allows speeding-up both read request messages (GETS L1 cache read-requests) and some write-related ones, i.e. invalidation messages, possibly multicast, and the corresponding acknowledgements. This implements a balanced approach to both read and write
operations initiated by L1 caches. Furthermore GETSs, despite belonging to a shorter critical path than writes in the coherency protocol evolution, can be more latency critical from the application point of view, as loads are more likely to stall the processor than stores. In addition, loads tend to be more than stores (e.g. about 75% vs 25%) and so, if the photonic path is not congested, GETSs can potentially improve the average network latency more than GETXs. With GETS photonic acceleration, canneal improves its execution time, scoring a 3% speedup. Now that GETX experience normal latency, race conditions are less than in the former case and canneal shows improvement due to GETS, Inv and Inv-Ack messages going through the low-latency path. Its all-to-all sharing pattern and the relatively high number of spin-locks (1.41% references against an average of about 0.29%) still induce a number of races due to invalidations. Some benchmarks, like blackscholes, fluidanimate, freqmine and x264, score a similar improvement as for GETX acceleration. dedup and vips reach 14% and 19% speedup, respectively. facesim and streamcluster now show about 3% improvement. This GETS+Inv+Inv/Ack configuration appears to be the most profitable among the tested ones. In fact, the fourth bars of Fig. 4b (GETX-GETS bars) show that when routing Inv, Inv-Ack and both GETX and GETS messages onto the photonic ring, performance are slightly degraded. In fact, results appear to algebraically summarize the advantages and disadvantages of both previous setups. The average speedup is of about 3% with a peak of 11% for blackscholes, vips, dedup and other benchmarks show slightly worse results than both the other configurations also due to the fact that the number of messages routed onto the photonic path start to be high enough to induce some queuing latency in the switches that inject traffic onto the photonic ring, thus eroding part of its intrinsic latency benefits. However, the other most important phenomenon that degrades performance in this case derives from the unbalanced usage of the electronic and photonic interconnection bandwidths. In fact this scheme, especially in conjunction with the specific access pattern (private, shared) of some benchmarks, and also due to the access distribution towards the different nodes, tend to privilege the photonic path for many control messages, thus incurring in some injection queuing even when the alternative electronic path is more free and would deliver the message in a shorter time. As future work we plan to investigate more in depth this issue and study the improvement of the present scheme with a dynamic selection policy in the switches that aims at choosing the electronic or photonic output ports according to the current link congestion and queue status. As a concluding remark, the comparison between Fig. 4a and Fig. 4b, highlights that some benchmarks benefit from the speedup of the considered network messages, while others (e.g. canneal) fail to translate a faster network into a reduction in execution time. In general, beyond the very specific features of canneal that we already commented on, our results show that the perceived performance of a complex multithreaded application are induced by the articulated interactions between the features of the overall on-chip network, the coherency protocol and the specific parallel behavior during execution and, in particular, by the adopted synchronization mechanisms. For this reason, such facets should be considered in a joint manner and, in the scope of the present work, we have identified a group of control messages that can induce a suite of complex multithreaded benchmarks to benefit from a simple low-latency photonic channel on-chip.

V. RELATED WORK

Over the last years, there has been a growing interest in photonic interconnection (ONoC) as a means to alleviate the bandwidth and power consumption problem of electronic networks. Several works have presented comparisons between on-chip electronic and photonic interconnects [11], [21], [22] in such sense. One attractive feature of photonic interconnection is the ability to operate in wavelength division multiplexing (WDM), which permits to transmit different independent wavelengths in the same waveguide at the same time, improving the maximum potential bandwidth. Some recent works have proposed complete photonic NoCs which rely on WDM capability [8]–[10], [23]. Vantrease et al [8] proposed a crossbar based photonic NoC with dense WDM for on-chip communication. Pan et al [10] proposed a cluster of nodes in which inter-cluster communication is done using similar crossbar architecture. Pasricha et al [9] proposed a 3D-stacked optical simple architecture that combines multiple photonic rings on various photonic layers with a 3D mesh NoC in active layers. Unlike the works discussed above, in this paper we propose to extend a traditional all-electronic mesh NoC with only one photonic ring. The simple ring waveguide reduces the overall network complexity compared to the cited works. Bahirat et al [23] also proposed an electronic mesh augmented by a simple photonic ring that is used to facilitate global on-chip communication between distant processors and memory cores, and sustains all the traffic (data and control). In our work, instead, the ring is used to provide a low-latency and high-bandwidth path to a specifically selected set of short but latency-critical control messages of the coherency protocol. Li et al [24] proposed a CMOS-compatible nanophotonic on-chip network with multi-layer design consisting of a WDM multicast subnetwork and a throughput-optimized circuit-switching nanophotonic subnetwork to optimize the latency-critical traffic and supports the circuit setup of circuit-switched communication. In such work, photonic signals travel over the air, relying on the reflection of light on a specific surface over the wafer itself. In our work we rely on a 3D stacked photonic ring to transmit the signal into silicon waveguides and we explore different latency-
critical message sets in a CMP scenario under multithreaded workloads.

VI. FUTURE WORK

ONoCs are known to be, by nature, more power efficient than the electronic counterparts, mainly because the power for transmitting signals is practically independent from the distance to be covered on-chip [11], [12]. In the next future we plan to include a power model in our proposal and study the power implications of selective traffic management in the hybrid electro-optical on-chip network, with the aim to identify the specific tradeoffs when both performance and power objectives are taken into account in a joint manner. Furthermore, enhanced CMP architectures, e.g. clustered, will be analyzed as to assess the stability of the proposed solutions over a range of architectural configurations as well their scalability to a higher number of processing cores.

VII. CONCLUSIONS

This paper presented an evaluation of the performance benefits of a hybrid electronic-optical on-chip network employing a simple 3D stacked optical ring for small, latency-critical, coherency control messages in a CMP scenario. The idea is to route a specific selection of control messages onto the photonic path as to reduce the overhead of the coherency protocol so that processing cores can communicate/collaborate/synchronize faster and, most importantly, without the complexity of a complete photonic network, which would be required to sustain the overall CMP traffic. Our results show that even a simple photonic structure, if the selected messages to inject are chosen properly, is able to boost the coherency protocol evolution significantly, and translate into a maximum 19% speedup in a specific benchmark and up to and average 6% across all the considered Parsec benchmarks. We have analyzed different groups of latency-critical control messages related to reads (GETS), writes (GETX), invalidations and acknowledgements. Among them, we identified that GETS, invalidations and invalidations acknowledgements is the best performing message mix on the considered benchmarks.

ACKNOWLEDGMENT

This work was partially supported by IT FIRB Photonica (RBFR08LE6V) and IT PRIN 2008 (200855LRP2) projects.

REFERENCES