Universidade Estadual de Campinas
Instituto de Computação

Jorge Luis Gonzalez Reaño

Photonics Opportunities in Modern Computing Systems

Oportunidades da Fotônica em Sistemas Computacionais Modernos

CAMPINAS
2021

Jorge Luis Gonzalez Reaño

Photonics Opportunities in Modern Computing Systems

Oportunidades da Fotônica em Sistemas Computacionais Modernos

Tese apresentada ao Instituto de Computação da Universidade Estadual de Campinas como parte dos requisitos para a obtenção do título de Doutor em Ciência da Computação.

Dissertation presented to the Institute of Computing of the University of Campinas in partial fulfillment of the requirements for the degree of Doctor in Computer Science.

Supervisor/Orientador: Prof. Dr. Rodolfo Jardim de Azevedo
Co-supervisor/Coorientador: Dr. Lois Orosa Nogueira

Este exemplar corresponde à versão final da Tese defendida por Jorge Luis Gonzalez Reaño e orientada pelo Prof. Dr. Rodolfo Jardim de Azevedo.

CAMPINAS
2021

Ficha catalográfica
Universidade Estadual de Campinas
Biblioteca do Instituto de Matemática, Estatística e Computação Científica
Ana Regina Machado - CRB 8/5467

G589p

Gonzalez Reaño, Jorge Luis, 1987-
Photonics opportunities in modern computing systems / Jorge Luis Gonzalez Reaño. – Campinas, SP : [s.n.], 2021.

Orientador: Rodolfo Jardim de Azevedo.
Coorientador: Lois Orosa Nogueira.
Tese (doutorado) – Universidade Estadual de Campinas, Instituto de Computação.

1. Arquitetura de computador. 2. Fotônica. 3. Sistemas de memória de computadores. 4. Memória desagregada. I. Azevedo, Rodolfo Jardim de, 1974-. II. Orosa Nogueira, Lois. III. Universidade Estadual de Campinas. Instituto de Computação. IV. Título.

Informações para Biblioteca Digital

Título em outro idioma: Oportunidades da fotônica em sistemas computacionais modernos
Palavras-chave em inglês:
Computer architecture
Photonics
Computer storage devices
Disaggregated memory
Área de concentração: Ciência da Computação
Titulação: Doutor em Ciência da Computação
Banca examinadora:
Rodolfo Jardim de Azevedo [Orientador]
Fabiano Passuelo Hessel
Paulo Sérgio Lopes de Souza
Hugo Enrique Hernandez Figueroa
Guido Costa Souza de Araújo
Data de defesa: 06-05-2021
Programa de Pós-Graduação: Ciência da Computação

Identificação e informações acadêmicas do(a) aluno(a)
- ORCID do autor: https://orcid.org/0000-0003-0828-8347
- Currículo Lattes do autor: http://lattes.cnpq.br/2341718150274711

Universidade Estadual de Campinas
Instituto de Computação

Jorge Luis Gonzalez Reaño

Photonics Opportunities in Modern Computing Systems

Oportunidades da Fotônica em Sistemas Computacionais Modernos

Banca Examinadora:

- Prof. Dr. Rodolfo Jardim de Azevedo
IC/UNICAMP
- Prof. Dr. Fabiano Passuelo Hessel
PUCRS
- Prof. Dr. Paulo Sérgio Lopes de Souza
ICMC/USP
- Prof. Dr. Hugo Enrique Hernandez Figueroa
FEEC/UNICAMP
- Prof. Dr. Guido Costa Souza de Araújo
IC/UNICAMP

A ata da defesa, assinada pelos membros da Comissão Examinadora, consta no SIGA/Sistema de Fluxo de Dissertação/Tese e na Secretaria do Programa da Unidade.

Campinas, 06 de maio de 2021

To Ruthy, Jorge and Gloria
- we walk in joy and difficulty.

Acknowledgements

Thank God for life and love.

To my beloved wife Ruthy; for your strength, kindness and love. Thank you, I love you.

I would like to thank my dear parents Jorge and Gloria, for their support and encouragement. Many thanks to my parents-in-law, Don Obidio and Dona Ruth.

I would like to thank my advisor, Professor Rodolfo Azevedo for providing me the opportunity to join his research group and continually supporting me.

I would also like to thank my co-advisor, Dr. Lois Orosa whose guidance was fundamental to this thesis's development.

I am very grateful to you both for your patience and academic guidance.

I would like to thank Professor Keren Bergman for allowing me to join her group at Columbia University.

To my dear friends: Aldo, Mary, Roger, Chrisnael, Robert and Rafael.

I would like to thank all who open me their heart and home (and gave me food): my IBBG family, Jessica and Samuel, Dona Elisia.

I would also like to extend my gratitude to my colleagues and friends from the LSC. Many thanks for the time shared during discussions, games, "pizzadas" and "cafezinhos". Obrigado!

This study was supported by the São Paulo Research Foundation (FAPESP) under grant 2014/01642-9.

Resumo

A fotônica emerge com dispositivos promissores para o projeto de sistemas de computação de última geração. Desenvolvimentos recentes nas técnicas de fabricação de fotônica de silício (SiP) mostram que a fotônica pode fornecer armazenamento óptico de dados, interconexões de alta largura de banda, comutação de alta velocidade e baixo *overhead* de energia-por-bit. O objetivo desta tese é explorar as novas oportunidades que a fotônica traz para a arquitetura de computadores: 1) avaliando as características e limitações de uma memória principal totalmente óptica, 2) propondo uma arquitetura óptica reconfigurável multi-GPU, e 3) propondo uma arquitetura de memória opticamente conectada para data centers desagregados.

Como resultado, apresentamos três contribuições. Primeiramente, propomos e avaliamos um processador com um sistema de memória totalmente óptica construído com uma topologia em árvore de switches em cascata para acessar as células de memória. O objetivo principal é obter baixa latência de acesso à memória, semelhante às latências dos caches L1 e L2. Esta característica é baseada na escala de operação $< ns$ de dispositivos fotônicos e permite eliminar os primeiros níveis de caches simplificando a hierarquia de memória, potencialmente economizando área de chip. Os resultados experimentais mostram até 37% de *speedup* com o benchmark SPEC2006 e até 50% com aplicativos de acesso à memória com padrão irregular, em comparação com um processador *single-core* tradicional com apenas um cache L1. A execução *multicore* com um cache de dados L1 reduzido de 2 KB mostra um *slowdown* médio de 23%.

Em segundo lugar, propomos e avaliamos o uso de interconexões ópticas com chaveamento para uma arquitetura multi-GPU. Usamos o NVIDIA NVLink comercial como base para nossa arquitetura óptica reconfigurável, pois é a plataforma preferida para os atuais aplicativos de aprendizado profundo. Implementamos uma versão simplificada do algoritmo de redirecionamento de largura de banda usado para realocação de tráfego em datacenters. Avaliamos nossa arquitetura proposta medindo o tráfego real da GPU e, em seguida, realizando uma análise estática de direcionamento da largura de banda. Nossos resultados mostram uma melhoria na taxa de dados de até 20% para modelos de redes neurais convolucionais bem conhecidos na literatura.

Finalmente, propomos e avaliamos uma arquitetura *Optically Connected Memory* (**OCM**) para desagregação da memória principal em data centers. Usamos ressonadores de micro-anel (MRRs) de última geração para projetar e avaliar no PhoenixSim os SiP links da OCM. Os links OCM podem sustentar a largura de banda DDR4 DRAM. Nossos resultados mostram um baixo *overhead* de energia-por-bit de 1.07 pJ equivalente a $\approx$ 10% da operação de energia DDR4. Para avaliar o desempenho com um simulador de nível de sistema, executamos benchmarks SPEC06, SPEC17, PARSEC, SPLASH e GAP de grafos. Comparado a um sistema sem memória desagregada, OCM mostra um *slowdown* médio de 17% com SPEC, e executa $\approx 1.3\times$ mais lento com PARSEC e SPLASH. OCM executa até $5.5\times$ mais rápido do que uma memória desagregada com conectores

40G PCIe NIC em nós de cómputo durante a execução de SPEC06. Avaliamos o OCM com um cache DRAM para reduzir o *overhead* de latência, mostrando uma desaceleração média de 38% com benchmarks SPEC17 e Grafos em comparação com um cenário sem desagregação. Acreditamos que o OCM é promissor para futuros data centers devido ao seu baixo consumo de energia e baixo *overhead* de latência.

Abstract

Photonics is emerging with promising devices for the design of next-gen computing systems. Recent developments in Silicon Photonics (SiP) fabrication techniques show that photonics can provide optical data-buffering, high-bandwidth interconnects, high-speed switching, and low energy-per-bit overhead. The objective of this thesis is to explore the new opportunities that photonics brings to computer architecture by: 1) evaluating the characteristics and limitations of a full-optical main memory, 2) proposing a reconfigurable optical multi-GPU architecture, and 3) proposing an optically connected memory architecture for disaggregated data centers. As a result, we present three contributions. First, we propose and evaluate a processor with a full-optical memory system built with a tree topology of cascaded switches for accessing the memory cells. The main goal is to obtain low memory access latency, similar to L1 and L2 caches latencies. This characteristic is based on the $< ns$ operation scale of photonic devices and enables eliminating the firsts levels of caches by simplifying the memory hierarchy, potentially saving chip area. Experimental results show up to 37% speedup with the SPEC2006 benchmark and up to 50% with irregular memory access pattern applications, comparing with a traditional single-core processor with only an L1 cache. Multicore execution with a reduced 2KB L1 data cache shows an average slowdown of 23%. Second, we propose and evaluate the usage of optical interconnects with switching for a multi-GPU architecture. We use the commercial NVIDIA NVLink as a basis for our reconfigurable optical architecture because it is the preferred platform for current deep learning applications. We implemented a simplified version of the bandwidth steering algorithm used for traffic reallocation in datacenters. We evaluate our proposed architecture by measuring real GPU traffic and then performing a statical bandwidth steering analysis. Our results show a data rate improvement up to 20% for well-known convolutional neural network models. Finally, we propose and evaluate an Optically Connected Memory (**OCM**) architecture for main memory disaggregation in data centers. We use state-of-the-art micro-ring resonators (MRRs) to design and evaluate in PhoenixSim the OCM SiP links. OCM links can sustain the DDR4 DRAM bandwidth. Our results show a low energy-per-bit overhead of 1.07 pJ equivalent to ≈10% of the DDR4 energy operation. To evaluate performance with a system-level simulator, we execute SPEC06, SPEC17, PARSEC, SPLASH, and GAP graph benchmarks. Compared to a system without disaggregated memory, OCM shows an average slowdown of 17% with SPEC and performs $\approx 1.3\times$ slower with PARSEC and SPLASH. OCM performs up to 5.5× faster than a disaggregated memory with 40G PCIe NIC connectors to computing nodes while executing SPEC06. We evaluate OCM with a DRAM cache to reduce the latency overhead, showing an average slowdown of 38% with SPEC17 and Graph benchmarks than a scenario without disaggregation. We believe OCM is promising for future data centers because of its low energy consumption and low latency overhead.

List of Figures

List of Tables

Contents

Chapter 1

Introduction

Photonic devices (also known as optical devices) are a relative newcomer for the computer architecture community because of their efficient interconnection capabilities and, the recently explored, data storage and processing potential. Silicon Photonics (SiP) is a semi-mature technology based on CMOS-compatible processes [92] already capable of high integration. Recently, hybrid silicon-compatible material platforms have been demonstrated, allowing the development of complex photonic integrated circuits (PIC) [250] with more efficient sources, photodetectors, modulators and other passive interconnecting devices.

The scalability of SiP has already been demonstrated. For example, similar to the first years of microelectronics, hundreds of optical multiplexing/demultiplexing devices can be integrated on a single chip. This points towards the beginning of a micro-photonics era, similar to Moore's law in the 70s [237, 235, 236].

Process Design Kits (PDK) with PIC are already offered by foundries, as CMOS compatible fabrication between photonics and electronics is already demonstrated [141, 83, 6]. However, it is expected that photonics reaches maturity in the following decades as hundreds of active devices (e.g., lasers) can be integrated monolithically with thousand of passive devices (e.g., photonic microring resonators arrays, multicore processors). Recent works in interposer development [5, 273], show significant advances for monolithic integration of passive and active photonics with electronic devices.

Photonic devices are widely used in sensor and telecommunication applications due to high data density transmission over large distances with low power consumption. These inherited characteristics by silicon-based photonic devices could address the bandwidth density and power limitations on computing systems.

As processing requirements increase with the application demands, prior proposals evaluated photonics as a high-bandwidth solution for communicating multiprocessors in networks-on-chip topologies [29, 226, 101]. Moreover, the imbalanced development between processor and memory technologies causes a performance gap referred to as the *memory wall.* Modern computing systems have a performance bottleneck when executing memory-intensive workloads of applications such as big-data analytics and machine learning. This problem, related to current memory technology limitations, shows that a memory system also requires higher bandwidth while scaling its capacity.

As photonic technology scale has already been demonstrated by integrating on the same die a RISC-V processor, optical interconnection and 1 MB memory bank; with ≈ 70 million transistors and 800 optical devices [241]. As a result, of the improved photonic device integration have been fabricated: a) volatile optical memory [199, 183], and b) non-volatile optical memory cells [212, 213].

An important open question is: what opportunities can photonics bring to the memory system? We seek to address this question by proposing and evaluating memory system architectures with optical devices in this work.

This thesis focuses on: 1) evaluating a full-optical memory system, 2) proposing an optical multi-GPU architecture, and 3) proposing an optically connected memory architecture for data centers. The first one analyzes fast memory operation, depending on the novel optical cells fabrication. At the same time, the latter two explores interconnection, which is more mature by providing high bandwidth with link reconfiguration mechanisms. Notice that despite optical processing (data operations) using photonic device meshes was already demonstrated [174], it is still in the early development stages and is out of this work scope. In [270], the authors presented an all-optical transistor, and in [115, 176], optical tensor hardware for machine learning was demonstrated.

Prior works focus on: i) showing photonic capabilities for computing systems [16, 159, 20, 232, 105, 274], or ii) introducing architectural mechanisms to support optical disaggregation [271, 8, 121, 177]. Similar to those works, we study how we can use photonics with processing elements such as multicore processors and GPU systems and evaluate its performance. This thesis also seeks to estimate the number of optical devices and their energy consumption when integrated into such a computing system.

This thesis is organized into five chapters. In Chapter 2, we describe the fundamentals and operation of photonic devices from a system perspective. We survey state-of-the-art devices and compare their key characteristics to help the computer architects outline their studies. In Chapter 3, we analyze the advantages and problems of optical data buffering for the main memory. Optical memories are seen as a potential cache replacement [158]. However, as photonics overcomes optical memory current limitations for massive integration, we must analyze their behavior compared to DRAM cells. Although their latency is in the picosecond scale, optical read and write operations are serialized because of the circuit switching nature and could lead to contention. We evaluate a L1 data cache reduction considering the optical read/write latency in ns. We perform a sensitivity analysis to estimate the cache reduction using a $x86\_64$ simulator in single-core and multicore scenarios, and PARSEC [37, 36] and SPEC06 [107] benchmarks. Based on our results, we consider that a full-optical memory system with a reduced 2K L1 data cache can perform similarly to a conventional system with 32 KB L1 (data and instruction) and L2 caches, being on average 23% slower.

In Chapter 4, we explore reconfigurable interconnection based on optical switches for a multi-Graphic Processing Unit (GPU) architecture [15]. We propose an architecture that relies on an optical circuit switch (OCS) to manage the *virtual* optical sublinks by wavelength filtering. Our study focuses on the bandwidth steering technique to reallocate bandwidth from GPUs underutilized links to GPUs that require higher transaction rates. We evaluate convolutional neural network execution on a data center server and perform a static analysis with the measured traffic during model training. In a multi-GPU scenario, the communication links are mainly used for direct memory access between the host and guest memory devices of processing elements. Our results show that an optically reconfigurable multi-GPU architecture can speed up to 20% compared to a conventional GPU server.

In Chapter 5, we explore how photonics can extend the memory capabilities towards exascale system [217]. We propose and evaluate the Optically Connected Memory (OCM) architecture for disaggregated datacenters. Our architecture benefits from the high-bandwidth and distance independent photonics characteristics to directly disaggregate main memory from the server's processing elements. We evaluate the performance of a multicore processor with performance executing PARSEC [36, 37], SPEC06 [107], SPEC17 [44], and Gapb [31] benchmarks. We estimate the number of photonic devices (i.e., microring resonators) for modulation/demodulation required to sustain modern DDR4 bandwidth. Our SiP link estimation uses the PhoenixSim simulator based on top-notch photonic device models. From our results, we observe that OCM produces a low energy-per-bit overhead of 1.07 pJ, and performs up to 5.5× faster than a disaggregated memory with conventional 40G PCIe network interfaces. In Chapter 6, we present our conclusions and discuss opportunities for future works.

Chapter 2

Background

Photonics is considered [221] the science that studies the generation, manipulation, amplification, transmission, modulation of photons. Its highly-integrable nature allows these functions to be realized with compact and energy-efficient optical devices. Moreover, Silicon photonics has been the key to solve the traditional photonics problems of integration and fabrication cost for more than 10 years now [249]. Its industry applications include mainly data center interconnects, and nowadays, high-performance computing for short-reach data communications.

This chapter presents a thorough analysis of the main components and devices responsible for realizing high-speed optical communications. We focus on the devices in the state-of-the-art that are compatible and can scale together with Complementary Metal Oxide Semiconductor (CMOS) fabrication processes.

2.1 Photonics, Optical Communications and Scaling

Optics is a solid contender to solve bandwidth and latency for high-performance computing interconnects through a semi-mature technology called Silicon Photonics (SiP). In addition, it allows maintaining or optimizing energy consumption. SiP is feasible by using developed CMOS processes [92], taking advantage of Silicon, which is a transparent material for the optical communications frequency band (from 1300 nm to 1650 nm)[224, 140] that allows for highly integrated photonic devices.

The possibility of using other silicon-compatible materials and define complex geometries has attracted much attention from the optics community in the later years [250]. The result is the possibility of developing complex Photonic Integrated Circuits (PIC). PICs enable breakthrough optical technologies ranging from high-bandwidth optical interconnects[58] to optical logic[208].

PICs for optical interconnects can provide high-aggregated bandwidth on an mm to cm size chip. They can provide miniaturized versions of matured optical fiber communications techniques with well-known physical dimensions for capacity scaling [259]: a) time, b) frequency, c) quadrature, d) polarization, and e) space.

2.2 Wavelength Division Multiplexing

Wavelength Division Multiplexing (WDM) is a technique that explores the physical dimension of frequency. Its main goal is to enable high aggregate bandwidth without adding more fibers to a single link, and it has been extremely successful for short-reach optical communication links. This technique has been widely used in optical fiber communications for almost three decades [202]. On the system level, WDM fiber links are characterized for being cost-effective and compact. It is important to note that fiber optic is the most expensive asset on a single WDM link, while transceiver components are comparatively cheap [259]. WDM has also become denser with the increased bandwidth-consuming devices and applications, counting more than a hundred channels per link.

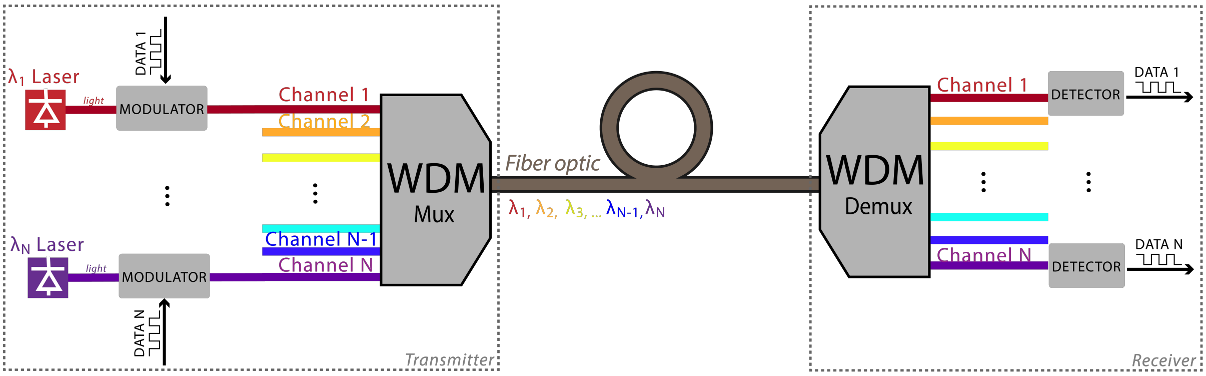


Figure 2.1: Simplified schematic of an unidirectional WDM link with N channels.

WDM is the multiplexing of many frequency channels on a single physical optical waveguide (mainly fiber optic). Fig. 2.1 shows a basic schematic with N channels, where the depicted main components constitute basic photonic devices. The devices that make up the transmitter and the receiver are usually integrated on a single chip [26]. The laser are usually comb lasers [260] which generate well-defined optical carrier in wavelengths λ_1, λ_2, ..., λ_{N-1}, λ_N. Each λ carries data after passing through a modulator. A modulated λ is called a channel. The fastest modulators are those driven by electric pulses known as electro-optic modulators. The period of the DATA signal determines the speed of the link. The most common modulators used for data center and HPC WDM interconnects are Microring Resonators (MRR) [26] silicon modulators.

MRR modulators can perform two main roles. One role is modulating light and multiplexing it onto a single optical waveguide. These devices offer highly compact WDM transceivers. The second role is that MRRs can also be used for demultiplexing the signal in the receiver end. After demultiplexing, each channel is photo detected, filtered, amplified, and interpreted to get each data set (DATA 1 through DATA N). Furthermore, a bidirectional WDM link has an RX and a TX on each end. Both transmitter and receiver with basic control have already been demonstrated integrated on a single chip [129, 272, 21] or co-packaged [196, 179, 254].

WDM links can scale up and build optical networks. Figure 2.2 shows a typical optical WDM network. A block for optical cross-connect devices is added to articulate and redirect individual WDM links λs to other parts of the network. We have purposely

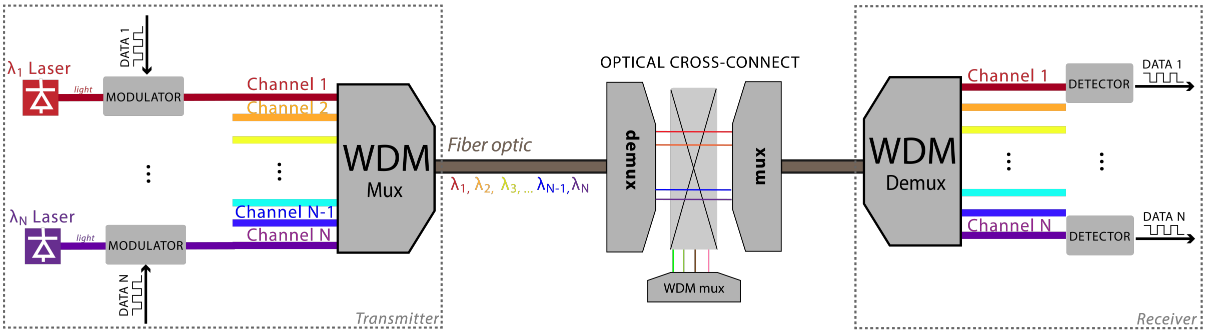


Figure 2.2: Simplified WDM network with switching capabilities.

left out amplifying elements in order to center our discussion on data center interconnects.

2.3 Photonic Devices

In this section, we discuss the main photonic devices for realizing a WDM link. For the transmitter end, light sources and modulators/multiplexors; for the receiver end, demultiplexers; for transmission, we present switches. We also include a discussion about optical memory cells for data storage.

2.3.1 Lasers

A previously noted (see Fig. 2.1) independent laser for each optical channel that generates continuous wave light (λ) is essential for realizing WDM. Initially, independent laser sources [180] were considered for WDM. Shortcomings included high power consumption and overall scaling problems. However, an increasing number of studies have shown the benefits of energy-efficient multi-wavelength light sources called Frequency Comb Generators (FCGs) [26, 52, 156].

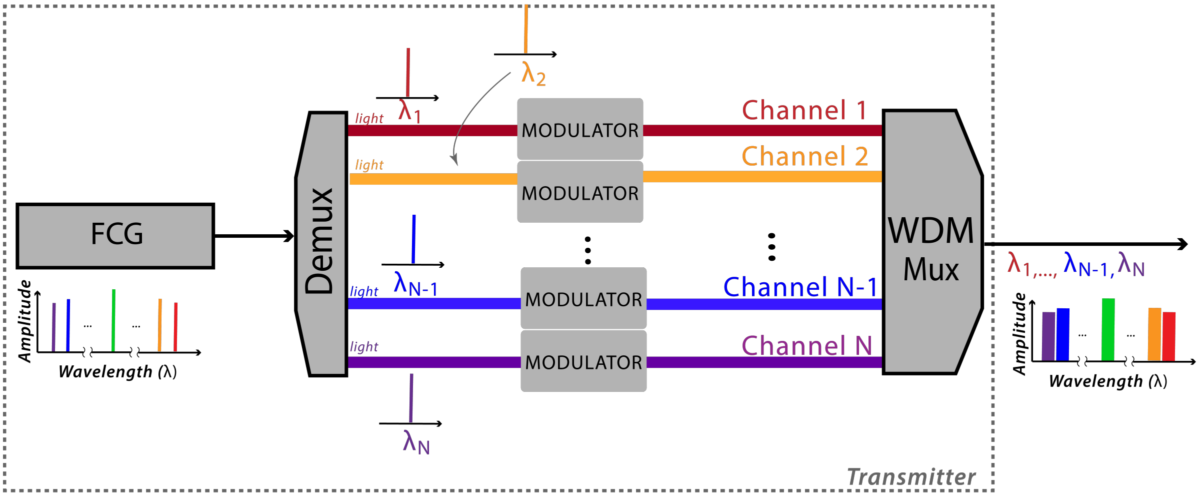


Figure 2.3: Simplified schematic of a WDM transmitter with a frequency comb generator (FCG).

FCGs are single devices that generate a number of equally-spaced optical carriers in the frequency domain (see FCG block in Fig. 2.3). A fundamental advantage of frequency combs is that comb lines are inherently equidistant in frequency [259], and they let the link designer avoid using a high number of independent continuous-wave lasers as previously depicted in Fig. 2.1.

In Figure 2.3, the FCG operates as a multi-wavelength source, then each wavelength (λ) is separated in a demultiplexer. After separation, each wavelength can be modulated with external electrical pulses in the modulator block. The modulated light is then multiplexed onto a single channel and can be coupled onto an optical fiber or interconnected to another section of the chip.

FCG-powered WDM links with an aggregate bandwidth that exceeds 1.2 Tbps [251] have been demonstrated in 2016. Following a rapid scaling, 179 optical channels have been demonstrated in 2017 [162]. Their work can transmit more than 50 Tbps. Furthermore, there are recent efforts for maximizing the number of comb lines (lambdas) in FCGs by using coupled photonic devices [46], which leads us to scale the number of channels and thus the aggregated bandwidth for more demanding applications. Recent works [90, 240] demonstrated FCG devices that require <1 W with an area of 1 cm^3.

2.3.2 Modulators

The main block for converting electric pulses into light is the electro-optical modulator. In Fig.2.4 light (λ_1) from a laser source (single lambda after demultiplexing of an FCG source can also be considered) is modulated by the serialized electric signal DATA 1. The electric pulses drive the MODULATOR block and modulate either the frequency, phase, polarization, or amplitude of light. The output channel contains all the information from DATA1 on optical frequencies (around λ_1), and its analog bandwidth (channel bandwidth in Fig.2.4) is delimited by the electrical model of the photonic devices that conform to the MODULATOR block. The analog bandwidth determines the channel's capacity in terms of the digital bandwidth (measured in bits-per-second). The aggregated capacity or bandwidth of a WDM link is the accumulated capacity of all individual channels.

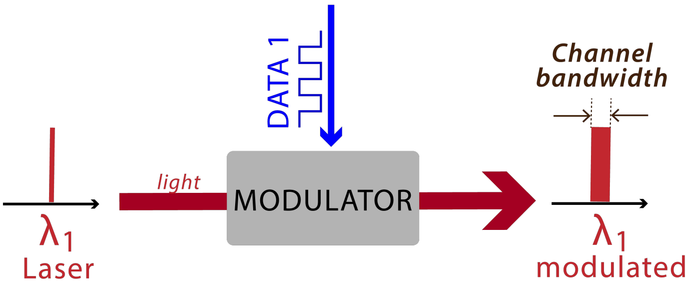


Figure 2.4: Simplified schematic of an electro-optic modulator. Electric and optical signals are represented in blue and red, respectively.

In Silicon Photonics, three out of the four fundamental blocks described in Fig.2.3, namely, demultiplexing, modulation, and WDM, can be realized on a single device such as an MRR modulator. The first demonstration of an MRR modulator was in 2005 [263], and it paved the way for Silicon Photonics in the industry. The possibility of small footprint, low-cost/high-scale production, and compatibility with CMOS processes proved that silicon could be a material to realize photonic integrated circuits. After 15 years of development, Silicon Photonics die market is in the order of 87-million-dollar[1].

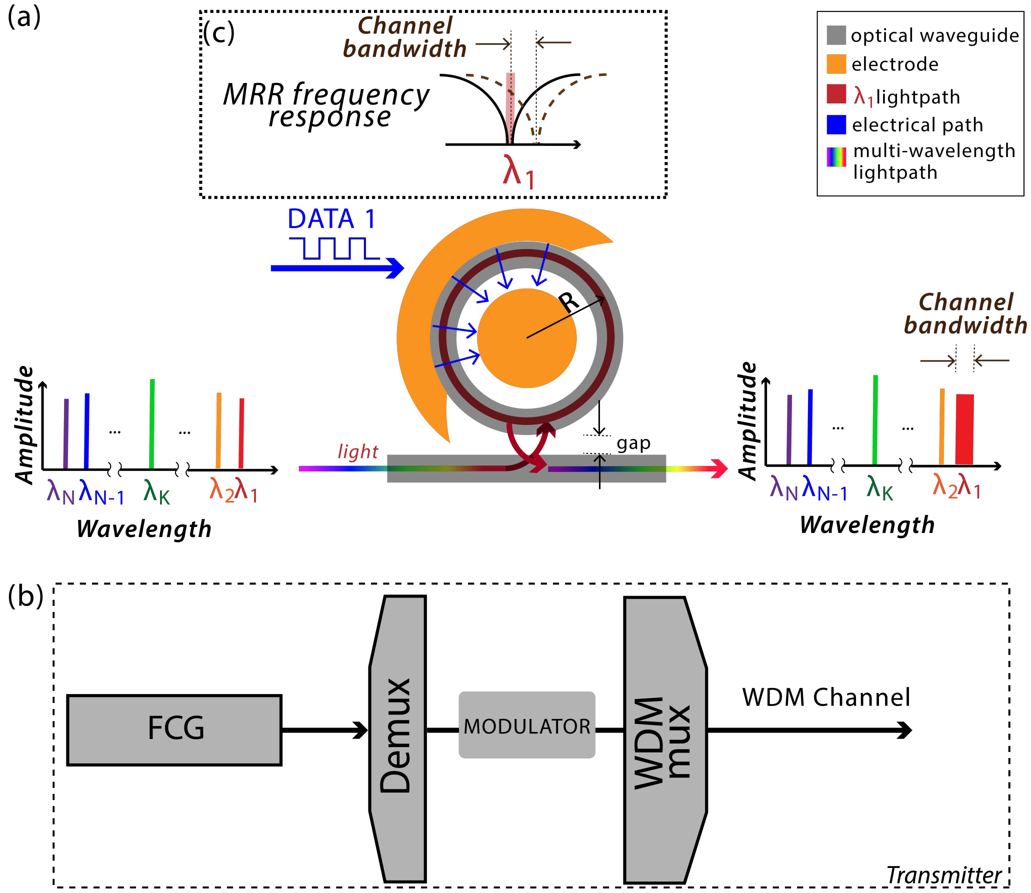


Figure 2.5: Two representations of a WDM single-channel modulator. (a) Simple schematic of microring modulator. Electric and optical paths are depicted in blue and red, respectively. In this MRR, R and gap are optimized to couple λ_1 light. (b) Block schematic of a single-modulator transmitter with frequency comb generator source. (c) The MRR modulator frequency response.

MRR modulators are optical waveguides[38] on a closed circular loop and can be classified with other resonant devices. As passive elements, they have a fixed radius and are depicted fairly close to a straight optical waveguide, as shown in Fig.2.5a. The gap determines the "strength" of light being coupled onto and from the ring. The radius determines the resonant lambdas. Thus, an MRR modulator can be designed to couple light from a single wavelength. As active elements, MRR modulators can use electric signals to control the phase of light using the plasma dispersion[209] effect that basically changes the properties of the material in optical frequencies by using energy from an external electrical source (blue lines in Fig. 2.5a). The frequency response of the MRR is a notch filter centered on λ_1. The modulation occurs when the electric field generated

[1] Yole Developpement Reports, 2020

by DATA1 affects the Silicon waveguide and shifts the resonance peak. This phenomenon generates a maximum (dashed brown plot in Fig 2.5(c)) that can be considered, for instance, a logical '1', and a logically complementary minimum (continuous black plot in Fig 2.5(c)) of the light intensity of λ_1.

The MRR modulators can demultiplex, modulate, and multiplex signals onto a WDM channel, as depicted in Fig. 2.5b). As a multi-wavelength signal passes through the straight optical waveguide, only a single wavelength λ_1 is selected and subsequently modulated by the electric signal in DATA 1. The MRR is electrically tuned to resonate only at wavelength λ_1.. After it resonates, the red light path (Fig. 2.5a) is re-coupled onto the straight waveguide and takes its place in the multi-wavelength signal spectrum.

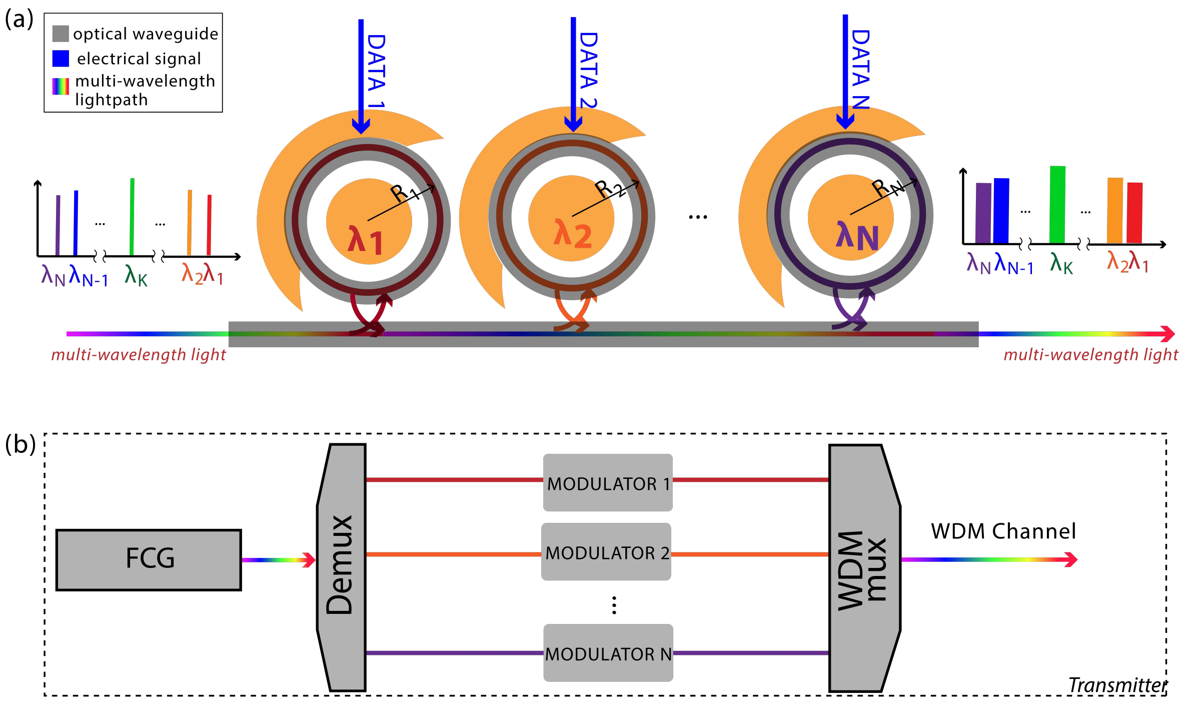


Figure 2.6: Two representations of a WDM modulator. (a) Simple schematic of MRR-based WDM modulator, and (b) block schematic of a WDM transmitter.

To match a real WDM transmitter, we can place tailored MRR modulators for each wavelength, as seen in Fig. 2.6a. Note that the individual channel bandwidth is important in a multi-MRR transmitter. Signal interference or cross-talk with any of its frequency neighbors can cause data loss. The result of a multi-MRR modulator is a multiplexed WDM channel with the desired aggregated bandwidth (N channels are aggregated in Fig. 2.6b).

A commonly used method to calculate the free spectral range (FSR) of the system assumes that each ring's frequency response is a notch filter centered on its respective λ. Then, FSR can be defined as the distance between two adjacent peaks of the WDM multiplexed signal[38]. As radii are specifically tailored for λ_i resonance, FSR directly relates to the WDM transmitter's on-chip area.

With improved fabrication techniques, state-of-the-art Silicon MRR modulators also improved modulation efficiency, energy, and area-on-chip. For instance, other electro-optic materials have been integrated with Silicon to improve modulation efficiency and lower driving voltage, such as silicon-organic[11] and silicon-LiNbO$_3$ [54] achieving femto-Joule (fJ) transmission. In terms of area-on-chip, plasmonic silicon modulators are up to 6 times smaller [103] than original MRR geometries, although they can be lossy in optical frequencies (around 2 to 3 dB per device).

2.3.3 Switches

The function of the cross-connect element in Fig. 2.2 is to switch information among many other fibers [66]. Wavelength multiplexers function is to combine several wavelength channels into a different aggregated channel. The switching capabilities on a high-speed optical network do not require O/E/O (optical-electrical-optical) conversion to switch information. A demultiplexer and a multiplexer are needed to select and switch specific channels in the optical domain and spatially cross-connect them. An optical switch is triggered with a low voltage or current.

Wavelength-selective switches are also based on MRRs. The frequency response of an MRR is the key to multiplex or demultiplex optical channels. The frequency response for a micro-ring resonator is a notch filter on the infrared regime. An MRR can work either: a) as a multiplexer coupled with one optical waveguide, or b) a demultiplexer coupled with two optical waveguides. As seen in Figure 2.7, there are two configurations for passive MRRs: all-pass, and add-drop. In the all-pass configuration (Fig.2.7a), only one lambda (λ_1) resonates in the ring and afterward is re-coupled onto the same optical bus or waveguide. R and gap are design parameters that determine the wavelength of resonance and the coupling strength. The add-drop configuration (Fig.2.7b) works as an add-drop filter; the resonant wavelength is dropped on to a second waveguide. All-pass MRRs are commonly used for modulators, while add-drop MRRs are commonly used for switch fabrics or cross-connects in optical networks.

It remains a challenge for future switches to exhibit: a) high-speed switching, b) low energy consumption, d) low area, d) wideband transmission, and e) low optical losses. Different than MRR's, there are other types of optical switches based on Mach-Zender Interferometers [206, 13], Microelectromechanical System (MEMS) [169, 75], plasmonic materials [248, 78] and Graphene [219]. However, MRRs switches are among the most promising devices showing selective wideband transmission with low energy and area, as demonstrated in [272, 59, 113] for multiport switches (e.g., 4×4 input/output ports).

Although there is a trade-off between critical characteristics, such as area and switching time, we compare several works on optical switches.

We compare several works on optical switches, although there is a trade-off between critical characteristics, e.g., area reduction does not imply fast switching. As shown in Table 2.1, the microring-resonator based devices exhibit a lower footprint and narrower bandwidth than MZI. While offering selective bandwidth filtering, they still offer higher bandwidth than plasmonic-based switches. With higher bandwidth, more wavelengths λ could be carried during propagation. However, MZI shows a higher area (mm^2) than plasmonic switches (μm^2), but the latter have higher losses during signal transmission. MRRs show broader bandwidth than MZI and plasmonic-based switches. For more information

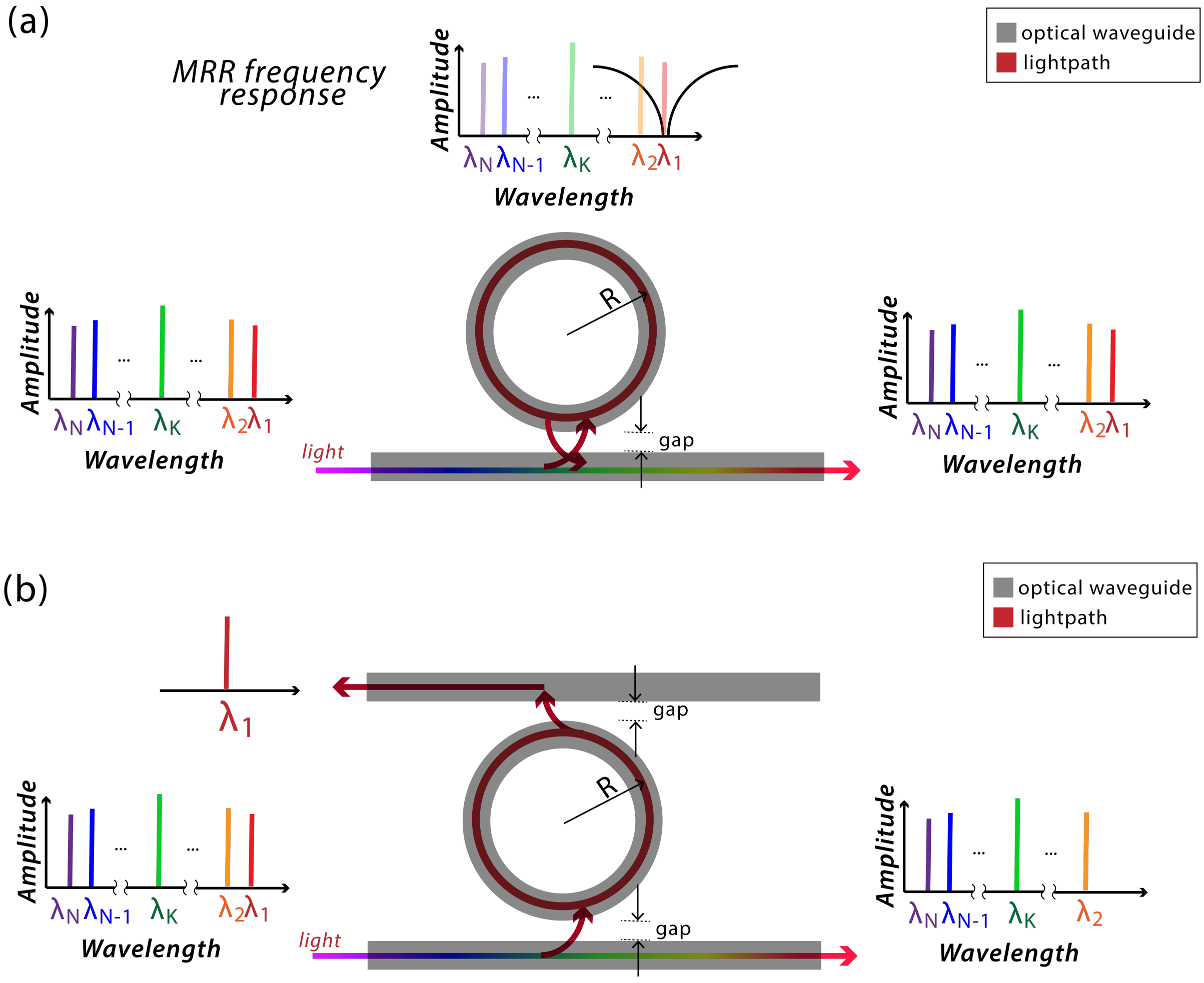


Figure 2.7: Optical micro-ring resonators. (a) All-pass configuration, and (b) Add-drop configuration.

about switches (e.g.: losses, crosstalk and timing), please refer to [246, 262].

Type	Works	Footprint	Switching	Control overhead	Losses	Energy
MZI	[206, 13, 128]	mm^2	ps	us	low	moderate
MRRs	[272, 59, 113]	10s μm^2	ps	us	moderate	moderate
Plasmonic	[248, 78]	100s nm^2	ps	us	high	low
MEM	[169, 75]	um^2	ns	us	low	high

Table 2.1: Average characteristics of optical switches. *Note:* These values represent an individual switch.

2.3.4 Memory Cells

Optical buffering is a very desirable functionality for both telecommunications and computer architecture. In this section, we will focus on the optical memory cell perspective related to computer architecture. For an extensive survey on optical memory, including optical packet storage, please refer to [10].

We observe that state-of-the-art optical memories can be classified into two main groups:

1. **Optical volatile memory cells:** There are two main types of optical volatile memory cells: 1) single-bit cross gained coupled cells, and 2) multi-bit photonic crystal with nanocavities memory cells.

    - Single-bit cross-gained coupled cells: use a passive photonic switch based on the Mach-Zehnder interferometer (MZI), and its behavior is similar to an SRAM cell. These devices split light into two separate paths (MZI arms) and join them after a specific electromagnetic path. Light interference can be constructive, partially constructive, or destructive, depending on the phase difference between the two MZI arms. This critical behavior enables the control of the amplitude of light, allowing for spatial switching. In several works [199, 158, 122], the authors used MZI-based switches to develop all-optical Flip-Flops (FFs) on a master-slave configuration, with low-energy and read, write functions. However, MZIs are known in the optical community for having a large footprint (see Table 2.1). There are efforts to shrink the size of these cells by using different materials and monolithic 3D approaches. Figure 2.8a depicts an optical FF composed of two active components, i.e., switches, in a master-slave topology. Each switch resonates at a different wavelength, where λ_a represents a logical 0 and λ_b a logical 1. Only one switch will be active per time, equivalent to a master state.

    - A multi-bit photonic crystal with nanocavities memory cell: is based on a periodic structure called photonic crystal. Depending on the relationship between the period of the structure and the wavelength, light can be slowed down, pushed forward, backward, or upwards (out-of-plane vector). Photonic crystal memory cells [132] use these properties to slow down the group velocity of light to a halt. While stored, the optical losses can be compensated with amplifying materials. However, the threshold for power compensation needs an external energy source (optical or electrical bias). The bias's amplitude has been demonstrated to relatively low, allowing for the whole device to work in pJ/bit. Figure 2.8b shows a photonic crystal memory cell that works under the injection lock principle. The structure has three nanocavities, each with a different operation wavelength λ_a, λ_b, and λ_c. An optical source traverses the memory cell where the injected wavelength has a specific amplitude threshold representing a low state or logical 0 and a high state or logical 1. The optical cell enters into a hysteresis loop, locking the state. For example, wavelength λ_a is in the low amplitude threshold, exhibiting the state for a logic 0.

2. **Optical non-volatile memory cells:** These [212, 175] are multiple-level optical memory cells with phase-change materials (PCM). Recent years have seen the emergence of optical PCM, potentially generating non-volatile photonic applications

Table 2.2: Optical memory cells evolution. Since 2001 optical memory cells show a trend in area reduction and increased energy efficiency.

Work	[151]	[109]	[110]	[178]	[152]	[199]	[268]	[122]	[149]	[51]	[183]	[132]	[197, 198]	[212, 239]	[85]	[213]
Year	2001	2001	2004	2006	2006	2008	2008	2010	2010	2011	2012	2014	2015	2015	2016	2019
Type	SOA ring	SOA MZI	micro ring	VCSEL	SOA MZI	SOA MZI	PHC nanocavities	VCSEL	micro disk	PhC nanolasers	BH-PhC nanocavities	PhC nanocavities	SOA MZI	PCM	PhC nanolasers	PCM nanocavities
Material	**	**	InP/InGaAsP	**	InP/SiO_2	**	InP/InGaAsP	InP/InAlGaAs	InP/SOI	InP/InGaAsP	InP/InGaAsP	InP/InGaAsP	InP	GST/Ge2Sb2Te5	InP/SOI	GST Ge2Sb2Te5
Area (μm^2)	1.3E13	**	720	36	5.4E6	**	<10	10	45	<10	<10	10	1.2E7	<250	6.2	<10
T (ps)	2.5E6	3	20	<1E3	200	<1E3	100	7	60	60	44	100	77	10E3	50	<1E3
E (fJ)	**	2E4	5	0.3	**	**	30	0.3	1.8	**	2.5	200	**	13.4E3	6.4	5E5
Volatile	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	No	Yes	No
Size (bits)	1	1	1	1	1	1	1	1	1	1	1	256	1	1(4 level)	1	1(13 level)

similar to the electronic PCM memories [238, 136]. The desired phase change for optical memories on the material is amorphous to crystalline while maintaining good transparency in optical communications wavelengths. Reading and writing functions can be performed with ultrashort optical pulses. Areas are the shortest among other kinds of memory cells. However, phase-change velocity can be slow, depending on the material. In [173], the authors analyze the state-of-the-art of PCM materials for optical memory devices, challenges, and future research direction. Figure 2.8c shows an optical memory cell with a PCM material, such as $Ge_2Sb_2Te_5$ (GST). The PCM material in the crystalline state attenuates the light propagation representing a logical 0, while the amorphous state represents a logical 1 because of the low attenuation. Considering that the PCM material can have intermediate amorphous states, it can operate as a multibit memory device.

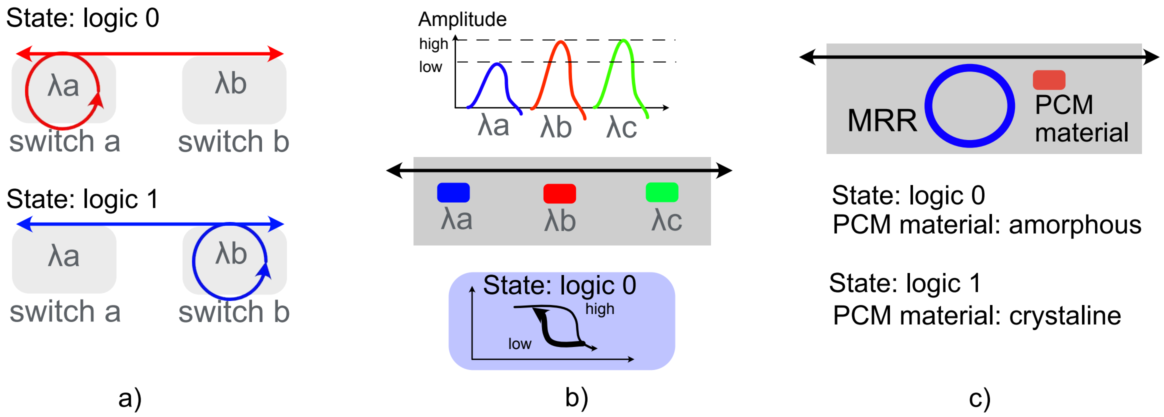


Figure 2.8: Optical memory cells operation, based on [10]. (a) set-reset optical flip-flop, (b) photonic crystal with nanocavities working with injection lock, and (c) Phase Change Material (PCM) optical cell.

2.4 Related Work

Pleros et al. [199] demonstrated a SOA-MZI based set-reset optical flip-flop (SR-FF). It consists of a cross-gain modulation (XGM) array of two coupled SOA-MZI, where each SOA-MZI device requires a 250 mA bias current. The reported footprint area is 45×12 mm^2. Vagionas et al. [252] discussed the theoretical analysis of the FF. Fitsios et al., in [86], measured the optical FF's operation speed of $\approx$ 75 ps.

In [158] the authors explored and evaluated a computer architecture with an 8 KB L1 optical cache architecture with an 8 core processor, using optical FF's as memory cells. They proposed an R-CAS (Raw-Column Address Selection) access to each optical memory cell, using a read/write λ-controlled mechanism. This mechanism was initially proposed by Kanellos et al. in [119]. Their multicore simulation envisions up to 256 KB L1 optical caches obtaining a 20% performance improvement over a conventional electrical two-level cached system running. The area and energy requirements are high because they used commercially available modules.

Pitris et al. [197] monotonically integrated the previously discussed SOA-MZI based FF in Indium phosphide (InP), where the area footprint is $\approx 12mm^2$. The SOA-MZI need a 170 mA bias current, and performs ps switching [86]. Pitris et al. [198] further integrated an access gate composed of two SOA's in order to enable the previously mentioned R-CAS mechanism. This access gate is responsible for enabling the set and reset FF signals via signal amplification. In Chapter 3, we study the usage of optical memory cells for the main memory system.

Brunina et al. [42, 43] introduce the concept of optically connected memory in a mesh topology connected with optical switches. Both works propose point-to-point direct access to the DRAM modules using Mach Zender modulators. These works motivate our study in optically connected memory. Brunina et al. [41] also experimentally demonstrate that microring modulators can be used for optically connecting DDR2 memory. Our study, in Chapter 5, builds on [41] to design the microring modulators used in our SiP links. There are several recent works [26, 229, 27] that propose analytical models of the microring used in our SiP links. Anderson et al. [16] extend the work of Brunina et al. [42, 43, 41] to experimentally demonstrate the optical switches using FPGAs for accessing memory.

These prior works [16, 41, 43, 42] are all experimental demonstrations to show photonic capabilities.

Table 2.3 shows related work for interconnecting memory and processing elements using photonics. This approach is currently known as disaggregation and is envisioned for data centers. A disaggregated data center comprises multiple resource pools, where photonics brings energy-efficient and reconfigurable interconnects [93] to: a) enable resource reallocation and b) avoid bottlenecks or underutilization. Disaggregation HP Enterprise initiative, known as The Machine [123], proposes a memory-centric system using photonics while improving the operating system layer for efficient memory allocation; also proposes a new programming model called Atlas. In [8, 271], the authors propose and evaluate the design of dRedBox based on optical switches focusing on the reconfiguration mechanism. In [82], the authors propose a GPU system with optical interconnects with a static traffic analysis based on simulation similar to our proposal from Chapter 4.

In contrast, this thesis Chapter 5 addresses three important questions for memory disaggregation that prior work does not: (1) How many optical devices (i.e., MRRs) do we need for current DDR technology? (Section 5.4.3), (2) What is the energy and area impact on the system? (Section 5.4.3), and (3) How does the processor interact with a disaggregated memory subsystem? (Section 5.4.2).

Some other works, such as [256, 275], point out, without system performance evaluation by executing real applications, that existing disaggregation protocols (i.e., PCIe and

Ethernet) could lead to high-performance loss. Our work uses system-level simulation to measure the performance overhead of such protocols. We propose to alleviate the optical serialization overhead by using the DDR protocol (Section 5.3.1). As photonic integration improves, we believe that the optical point-to-point links will become the main candidates for interconnecting disaggregated memory. With our PhoenixSim [216] model, we explore the design of SiP links based on DDR requirements. Our proposal can be used to improve existing PCIe+photonics works, such as [264].

Works	System-level simulation	Optical link design/eval.	CPU	GPU
[41, 274]	✗	✓	✓	✗
[8]	✓	✓	✓	✗
[123]	✗	✓	✓	✗
[82]	✓	✗	✗	✓

Table 2.3: Related works with optical disaggregation of memory and processing elements in future data centers.

Yan et al. [264] propose a PCIe Switch and Interface Card (SIC) to replace Network Interface Cards (NIC) for disaggregation. SIC is composed of commercial optical devices and is capable of interconnecting server blades in disaggregated data centers. The evaluated SIC shows a total roundtrip latency up to 426 ns. In contrast, the scope of our work is point-to-point DDR DRAM disaggregation without PCIe or other additional protocols.

Other related disaggregated memory prior works (1) explore silicon photonics integration with a many-core chip in an optical network-on-chip design [30], (2) propose the design of a DRAM chip with photonic inter-bank communication [32], (3) present an optoelectronic chip for communication in disaggregated systems with 4-λ and an energy consumption of 3.4 pJ/bit [7], (4) evaluate a memory disaggregation architecture with optical switches focusing on re-allocation mechanisms [271], (5) analyze the cost viability of optical memory disaggregation [4], and (6) evaluate memory disaggregation using software mechanisms with high latency penalties in the order of μs [99]. Unlike [271, 7, 32, 4, 99], our study evaluates i) system performance with real applications, ii) the design of the SiP link for DDR DRAM requirements, and iii) SiP link energy for a disaggregated memory system.

2.5 Final Considerations

This chapter detailed the Silicon Photonics (SiP) operation, including optical modulators, muxes, switches, and memory cells. Recent works show a positive trend in photonic device integration, reducing the device footprint, and increasing feasibility for large device fabrication. Tables 2.1 and 2.2 summarizes the main characteristics of switches and optical memory cells. We observe that computer architects could benefit from the direct classification of SiP device characteristics to simplify the system design. This chapter detailed the Silicon Photonics (SiP) operation, including optical modulators, muxes, switches, and memory cells. Recent works [80, 94] show a positive trend in photonic device integration, reducing the device footprint, and increasing feasibility for large device fabrication. Tables 2.1 and 2.2 summarizes the main characteristics of switches and optical memory cells, which can be used as building blocks. We observe that computer architects could benefit from the direct classification of SiP device characteristics to simplify the system design. As photonic fabrication matures, essential metrics such as the experimental control delay for massive integrated optical switches or memory cells optical will become available.

Chapter 3

Full-optical Memory

This chapter explores a full-optical memory system's key design considerations, using state-of-the-art optical devices with a wavelength routing method. In 2016 [96], we propose a large implementation of a MB order memory and evaluate the size reduction for the cache hierarchy. This chapter discusses challenges for a full-memory architecture and evaluates the limitations on the photonic interconnect. We show the potential better performance compared to a conventional electrical system in a single core and multicore scenario. It is crucial to notice that such massive architecture is far from fabrication maturity because of current characteristics of optical switches: 1) low count integration, and 2) high latencies imposed by WDM control.

3.1 Photonics for the Main Memory System

Photonics have been used successfully in long link communications for many years due to its intrinsic characteristics such as distance-independent low-power dissipation, high communication bandwidth and low crosstalk. Design space exploration for optical on-chip devices is an emergent research topic in computer architecture. Initial research direction pointed to Optical Network-on-Chip (ONoC), which uses wavelength routing mechanisms to communicate multiple cores within a chip under different topologies and arbitration schemes [29, 226, 101].

A roadmap from the industry and the research community [94] highlights a clear path for on-chip silicon photonics interconnections, pointing out the promising scalability expectations in the next two decades to achieve a full-optical device integration in a single chip. Although optical fabrication technology is still not mature, several small-scale integrated silicon photonic devices have been recently fabricated [250]. The monolithic integration of silicon photonics modules with processors become feasible [241] due to those recent advances in device fabrication.

Another emerging photonic research topic is optical memory fabrication. Optical memories leverage different techniques to achieve data buffering in optical memory cells. To date, a 1-bit memory cell can be implemented as a Semiconductor Optical Amplifier (SOA) based flip-flop [199], and a wider cell is feasible using emergent nanocavity technologies [132]. Also optical Phase-Change Memories (PCM) were demostrated with non-volatile multibit storage capabilities [212].

In [96], we explored a new architecture for a single core processor with an optical RAM (o-RAM) that could overcome the limitations of current electrical memory subsystems. We focused on the co-design of processors and these memories. Unlike other approaches using an ONoC and electrical memory, our proposal does not require optoelectrical conversion on the memory side, reducing its overhead. Furthermore, the o-RAM operation latency is in the order of picoseconds (the carrier is in THz).

Our exploration of a full-optical memory is motivated by the growing disparity between the performance of current DRAM technology and the processor data access requirements. Despite being in the early development stage, we consider that optical memory could be an alternative for future computing systems similar to current relatively mature electrical memory devices such as PCM and memristors. The two main characteristics of our full-optical memory architecture are:

1. **Using the O-RAM as the main memory system.** Where the optical devices not only offers bandwidth benefits for interconnection but also for high-speed data storage.

2. **Cache hierarchy reduction.** We considered a reduction in the cache hierarchy by estimation of the optical memory access. If the ORAM latency is equal to or lower than the last-level cache latency, then we could remove this hierarchy level or reduce its size.

The following section addresses how to architect an optical RAM (o-RAM) of MB size. First, we detail the high-level architecture with its communication link based on state-of-the-art photonic devices such as transceivers, switches, and memory cells. Second, we evaluate our memory system proposal using microarchitectural execution-driven simulation. Our exploration indicates o-RAM low operation latencies between the values of the current first-level caches. As a result, we obtain speedup compared to a conventional electric system due to the low latency for memory operations. Finally, we discuss the main challenges for architecting a full-optical memory.

3.1.1 Architecture Overview

Figure 3.1 shows our architecture for a single core with an optical memory [96]. Our proposal sets a direct link from processor to memory due to circuit switching nature of optical networks, also aims to reduce the number of electro-optical conversions during communication. Based on an optical communication scheme, there are three blocks: 1) a transceiver based on modulators/demodulators [38], 2) an optical network (ONoC) formed by switches based on MRRs [59] or MZI[128], and 3) optical memory cells as an optical memory bank based on multibit volatile cells [132] or non-volatile cells [212].

Our ONoC design relies on a tree topology to perform direct access to all memory cells by routing the light beam, where the tree branches are cascaded optical switches (o-SW). Figure 3.2 shows the o-SW topology as a complete binary tree graph where the nodes are the 1 × 2 o-SW, and its control relies on electrical signals. The number of memory cells is equal to the number of the tree leaves, and with this configuration. Figure 3.2 shows

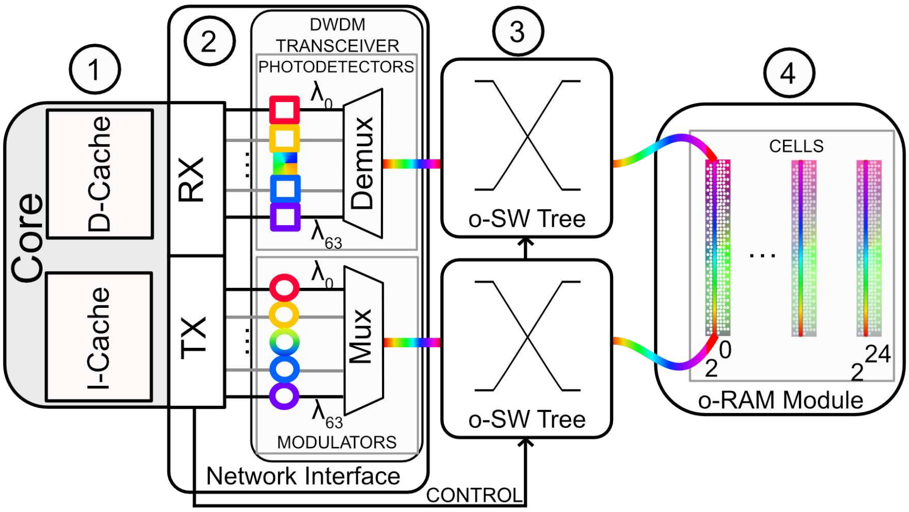


Figure 3.1: Overview of a processor with an o-RAM system model: 1) a single core with L1 caches, 2) a DWDM transceiver based network interface, 3) an optical link based on o-SW and 4) an o-RAM bank.

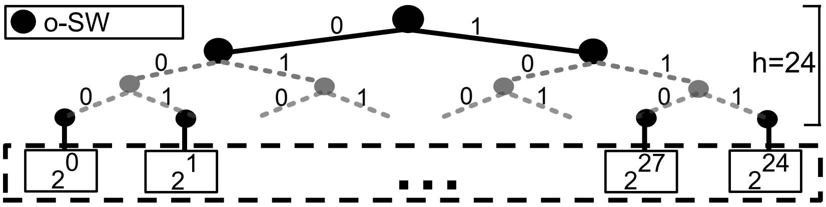


Figure 3.2: Optical switch (o-SW) topology for an 256 MB o-RAM.

the ONoC for a total of 256 MB using 256-bit (32 B) volatile memory cells from [132], therefore the binary tree has *height* = 24, and its total number of switches is (2^{24-1}).

A processor ❶ communicates with the o-RAM using an optical communication interface composed by the processor Transmission (Tx) and Receiver (Rx) modules, and the DWDM transceiver (Figure 3.1 ❷). Each Tx's and Rx's pin is directly modulated or demodulated with the DWDM transceiver setting a high-bit-rate data link.

The physical link is composed by two o-SW trees as shown in Figure 3.1 ❸, which are controlled to establish a direct access to the o-RAM by the optical communication interface. Due to the system circuit switching nature, the interface can only handle one in-flight request (serialized access). Therefore, o-RAM access could cause contention, increasing the memory latency. Then, read/write instructions would have to stall until previous o-RAM operations have finished. One way to minimize the contention is to increase the number of paths to each o-RAM cell, at the cost of higher area footprint. The set of paths to an o-RAM cell defines the total number of o-RAM ports. For example, in Figure 3.1 ❹ an o-RAM cell has a single path. For an o-RAM implementation with two ports, we need to duplicate the number of interfaces and o-SW trees structures. If we consider an ideal MRRs of 5 $\mu m^2 \times (2^{24} - 1)$, a single o-SW plane will have an area of ≈0.83 cm^2. Then, doubling the number of paths is ≈1.7 cm^2.

There are two o-SW trees in our single port design, one before the o-RAM cell to perform write operations, and the other after for read operations. When a write operation is performed a path is set from the processor Tx to the o-RAM, modulating the data on the initiator and storing it on the target o-RAM cell structure. However, for a read operation, a closed loop is established. This is because a path is set using both o-SW trees, one from processor Tx to o-RAM, and another from o-RAM to processor Rx.

O-RAM effective read/write latencies are one order of magnitude lower than conventional electrical DRAMs and in the same order as on-chip caches, without WDM control and modulation latencies. This allowed us to reassess the memory hierarchy of electrical caches with equal or higher latencies. We propose to only use L1 cache with the o-RAM, this could reduce the processor area since the L2 and L3 cache can be approximately one-half of the die's silicon area [220].

Fig. 3.3 shows a multicore system with L1 Data (L1D) and Instruction (L1I) caches without further levels of cache hierarchy. We called this L1D cache as a data buffer because its size is lower than a conventional L1D in modern proccessors. In our design, the L1I cache is electrical to maintain the instruction lines available to the processor, avoiding excessive stall cycles due to misses for instruction fetching.

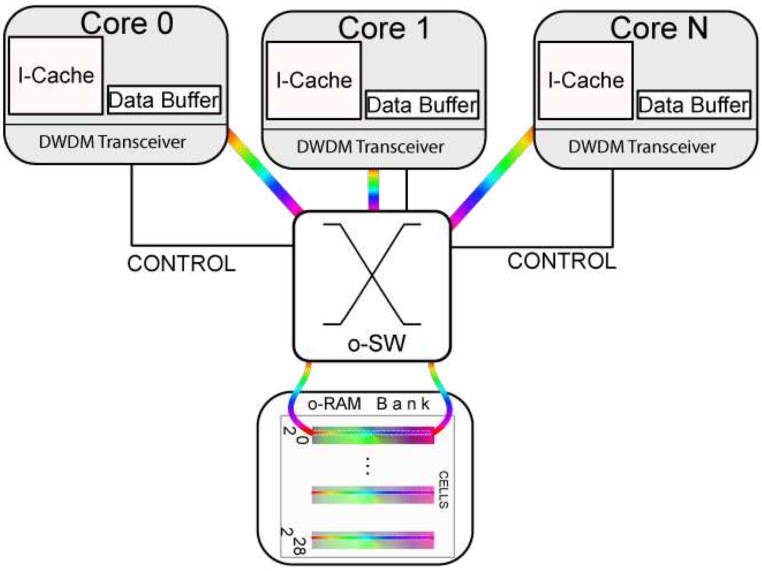


Figure 3.3: Multicore with an full-optical RAM. The processor only have one level of cache, where the Data cache size can be equal or smaller than the Instruction cache size.

3.1.2 Operation and Timing

The optical memory operations could be summarized as follows:

- **WRITE** For writing data on an optical bank, the processor output is modulated in the transceiver; then, a path is set within the optical network using electrical or optical control signals depending on the switch characteristics. The optical signal travels on the optical network to enter the memory cells and finally stay optically buffered.

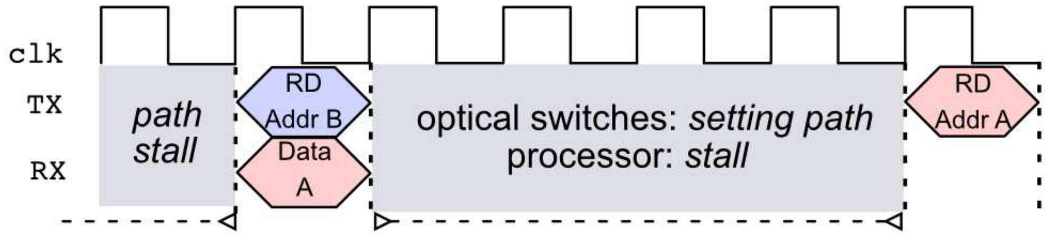


Figure 3.4: Optical memory system read/write timing diagram.

- **READ** For reading data from an optical memory bank, the processor transceiver modulates the electrical signals and sets the path using electrical or optical control signals on the switches. The optical signal passes through the optical network and reaches the optical bank. A read optical signal needs higher signal amplitude than a write optical signal because the signal will traverse the memory cells and pass backward the optical network to the transceiver's demodulators suffering more attenuation.

In our timing model, the overall total latency is defined by the optical switching latency that defines the setting path time. It does not consider other latencies caused by the electrical activation of optical switches or electrical control of modulators. This approach is similar to other works, such as [226, 32, 30, 161]. It is motivated by two key factors: 1) it envisions a chip with fully integrated photonic capabilities that can be placed in a computing system, then latencies such as modulation tuning and serialization need to be minimum due to the high efficiency of specialized hardware, and 2) other latencies, such as group propagation in fiber, are negligible because of the on-chip or on-board distance (max., in the order of tens of centimeters). For an extensive model, please refer to Chapter 5, which details a disaggregated system (meters of rack distance) using state-of-the-art optical devices, and we evaluate all the latencies.

Figure 3.4 shows the timing diagram for the read and write operations assuming a 2 GHz processor clock, a memory operation latency below one processor cycle, and a switching delay of 2 ns (4 cycles). The switching delay is an optimistic assumption. For example, although the effective switching latency is in ps for MRR's optical switches, they operate through thermal effect, leading to μs order latency [24, 16]. In Section 3.2, we discuss recent work for fast optical switching.

In an ideal scenario, the total o-RAM access latency is 7 cycles. In cycle 1, the processor performs a request to memory, and the communication interface modifies the optical switch states to set the path to the requested data address (Addr). Cycle 2 contains the decoding address delay. The communication interface sets the path in 4 cycles (from cycle 3 to 6), as all the switches are activated simultaneously in a 2 ns electrical stimuli. In cycle 7, as a result of the optical cells' picoscale operation, the data is modulated with the transceiver and stored in the memory cell in the case of a store. Otherwise, the memory cell outputs the data stored for memory reads, which moves the data to an electrical buffer or processor cache.

3.2 Discussion of Implementation Challenges

We identified three key challenges to achieve a full-optical memory implementation, all related to the improvement of photonic device fabrication and design as technology matures:

1. **Provide fast switching:** Reported MZI and MRRs switches exhibit optical switching delay in the order of ns for wavelength selection, enabling low access time for our o-RAM design. However, as reported [24, 16], the latency caused by the thermal effect required for control (i.e., optical switch phase shift) is up to 5 μs order latency. The design of efficient optical switches is an active research area, as shown in [113, 200]. Recent works reported total switching delay: 1) in the order of 200-400 ns at the cost of a narrower bandwidth as in [116, 160, 203], and 2) in the order of fs using the nonlinear effects with plasmonics or photonics crystals in [190]. Recent demonstrations in [77, 76] of optical devices, based on the Pockels effect, show promising results for high-performance switches and modulators. This type of electro-optic switch operates at cryogenic temperatures and does not rely on the Joule heating effect, obtaining efficient optical switches with high bandwidth, low loss, and high-speed. However, their operation is dependant on the Pockels properties of three main materials used in thin films [165]: a) lithium niobate (LiNbO3), b) lead zirconate titanate (PZT), and c) barium titanate (BaTiO3 or BTO). We observe that fast optical switching can be achievable in future optical switches as material applications mature.

2. **Improve the integration feasibility:** Future photonics chips need to integrate thousands of photonic devices, being two orders of magnitude higher than the state-of-the-art devices such as the switch arrays from [60] where dozens of switches are used to form Benes topologies, e.g., 16 × 16 or 32 × 32 topologies. Notice that it remains a challenge the massive integration, as we propose using millions of o-sw. It is expected to achieve such high integration during the end of the next two decades, according to The Integrated Photonic Systems Roadmap-International (IPSR-I) [94]. It remains a challenge the massive integration of nonlinear photonic devices, such as the multi-bit optical memory cell [132], because: a) most of them are not compatible for integration with active photonic devices (e.g., lasers, modulators), and b) is still challenging to maintain high reproducibility of the nonlinear behavior for massive devices. Please refer to [142] for a discussion and analysis of nonlinear photonics integration with Silicon devices.

   Recent work using plasmonic materials shows promising results for area reduction of optical switches due to its high integration and compatibility with CMOS processes [190]. For example, an MZI [253] o-sw has an area of $\approx 0.02\ mm^2$, then the o-sw tree area in our design will be 0.3 m^2. Using plasmonic o-sw [265], each with 4.8 μm^2, the total required area will be 80.5 mm^2. In [79], the authors fabricated an atomic scale plasmonic switch, enabling research for future low footprint o-sw devices. We observe a clear trend in area reduction of optical switches during the last decade. However, massive integration of photonic devices matures with the fabrication processes.

3. **Reduce the optical losses:** Our o-sw tree design relies on cascaded optical switches, which cause a loss in the transmitted signal. For example, for a 24 height o-sw tree[1] using plasmonic o-sw [265] the optical losses are up to 57.6 dB, and using MZI [253] o-sw up to 69.6 dB. Figure 3.5 shows an state-of-the-art MRR optical switch simulation using the reported parameters from [60]. As shown in Figure 3.6, the estimated cross loss is 1.75 dB and the bar loss is 1.2 dB, equivalent to an optical loss of 42 dB for a 24 height tree.

   For our system implementation with this MRR switch, we need intermediate stages of amplification between switching levels of the o-sw tree to guarantee the optical signal propagation, significantly increasing the total design footprint.

   It is essential to notice that MRRs and MZI switches offer lower losses than plasmonic switches. Techniques and methods for obtaining reasonable energy levels and low losses with plasmonic materials are open issues that have gained considerable attention from researchers [125].

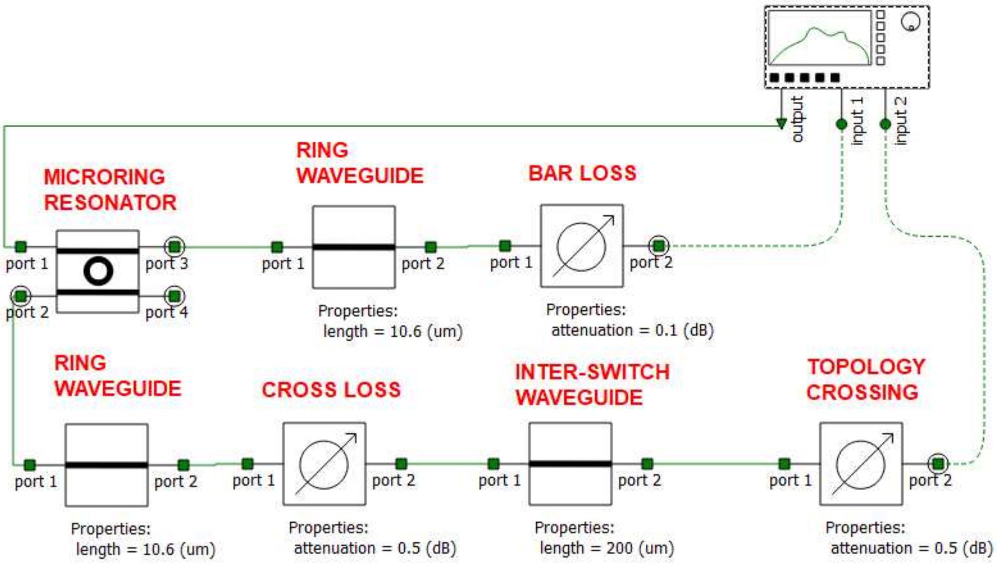


Figure 3.5: Optical switch (o-SW) simulation based on the measured parameters from [60].

We expect that future device fabrication techniques will alleviate the previously discussed trade-offs. While there are relevant technology challenges, we conclude that implementing such a massive optical RAM is still not feasible and is still far from maturity. Our goal is to show the potential impact of photonics and motivate further studies at the microarchitectural level to define requirements and limitations at the physical level.

[1] Without considering timing and area footprint

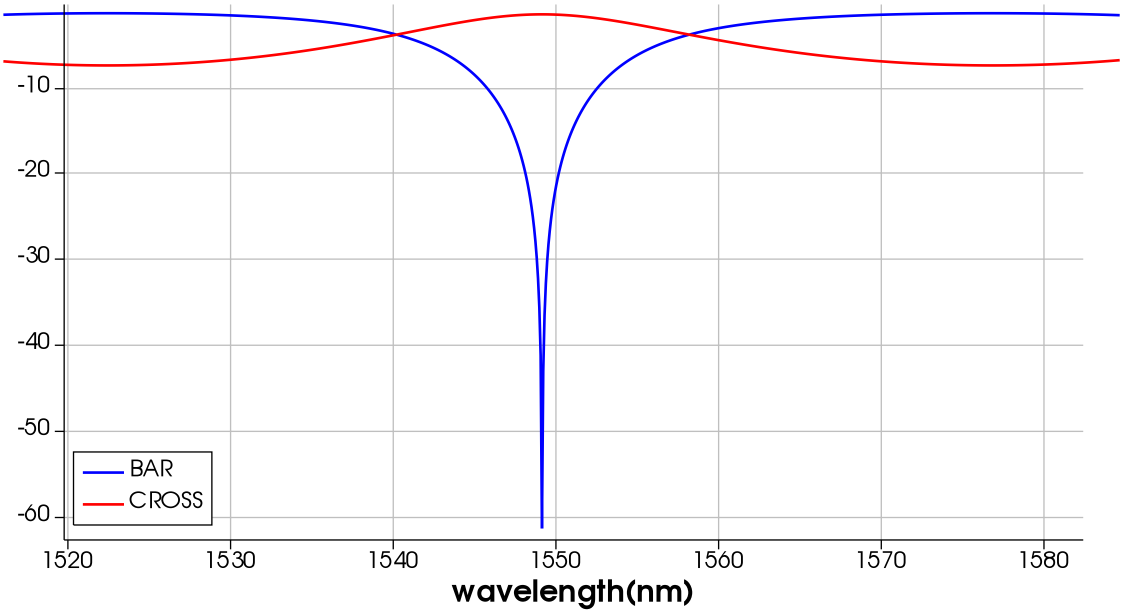


Figure 3.6: Optical switch (o-SW) loss results using the measured parameters from [60].

3.3 Evaluation and Results

To assess a processor's performance with o-RAM architecture, we used a modified version of ZSIM simulator [222]. Table 3.1 details the three main benchmarks used in our evaluation: 1) we used SPEC2006 [107] with the Simpoint [195] methodology, 2) we executed PARSEC [36, 37] with medium inputs, and 3) We also executed a set or irregular applications [153] which includes a Page Rank algorithm implementation and a Random Memory Access application. Our architecture relies on two key characteristics: a) multibit o-RAM storage (32 B), and b) fast-switching (tens of nanoseconds, see Section 2.3.4 for MZI latencies), which is an ideal scenario without WDM imposed latencies (see Section 5.3.3 for a complete evaluation of a WDM link).

SPEC2006 [107] (*pinballs*)	cactusADM, hmmer, astar, calculix, leslie3d, soplex, bwaves, dealII, mcf, sphinx3, gobmk, milc, tonto, gromacs, gcc, GemsFDTD, h264ref, perlbench, zeusmp, povray,amd, wrf, bzip2, gamess, omnetpp, xalancbmk
Irregular [153]	pagerank, random memory access
PARSEC [37, 36] (*medium*)	raytrace, canneal, fluidanimate, blackscholes, splash2x.fft, splash2x.barnes, splash2x.oceancp, splash2x.waternsquared, splash2x.waterspatial, splash2x.radiosity, splash2x.cholesky, splash2x.radix

Table 3.1: Benchmarks

We evaluated two scenarios for the optical memory system, with our o-RAM in ZSIM that models the contention and circuit switching for different memory cell access in our architecture.

Scenario 1: Single core with only L1 cache

Table 3.2 summarizes our setup configuration and details the main characteristics of an electrical processor system with: a) DRAM and, b) an optical memory system (o-RAM). On both electrical and optical architectures, we use a 2GHz `x86_64` processor model. In the electrical platform, we use an L1+L2 cache with DDR3-1066-CL8 memory. We defined three latencies values: 5, 7, and 11 cycles, based on average switching delays reported values of optical switches [253, 61] without considering tuning delays. We performed a sensitivity analysis for the o-RAM evaluation, defining the number of o-RAM banks (M) and ports (P) per bank as a power of two and up to 4.

	Processor	
Cores	1 (x86_64)	
Frequency	2 GHz	
	Main Memory	
	Baseline	**Optical**
Type	DDR3-1066	o-RAM
Latency (cycles)	150	5, 7, 11
	Cache	
Levels	L1 I+D (64 KB) L2 (2 MB)	L1 I+D (64KB)
Latency (cycles)	L1 : 4	L1: 4
Associativity	L1: 4-way L2: 16-way	L1: 4-way
Block Size	32 B	
Replacement	LRU	

Table 3.2: Configuration to evaluate the scenario with o-RAM and a single level cache

Figure 3.7 reports the geometric mean speedup with SPEC2006 and irregular applications. Results are grouped by the number of modeled ports (P) and memory banks (M) and normalized to the electrical baseline platform detailed previously in this section. Each bar has three levels for the 11, 7, and 5 cycles access latencies evaluated, where the first bar shows the results with SPEC2006, and the other bar was obtained with irregular applications. Bars (A) and (B) are the results of a system with an L1 I-cache and D-cache with an o-RAM.

All cases obtained better performance than the electrical case because of its lower access latency. The speedup is up to 38% with SPEC2006 and 84% with irregular applications. Furthermore, our experiments show promising results with irregular applications. As detailed in Fig. 3.7, the o-RAM system obtained an speedup up to 84% (B).

The speedup is higher when the memory level parallelism increases due to a higher

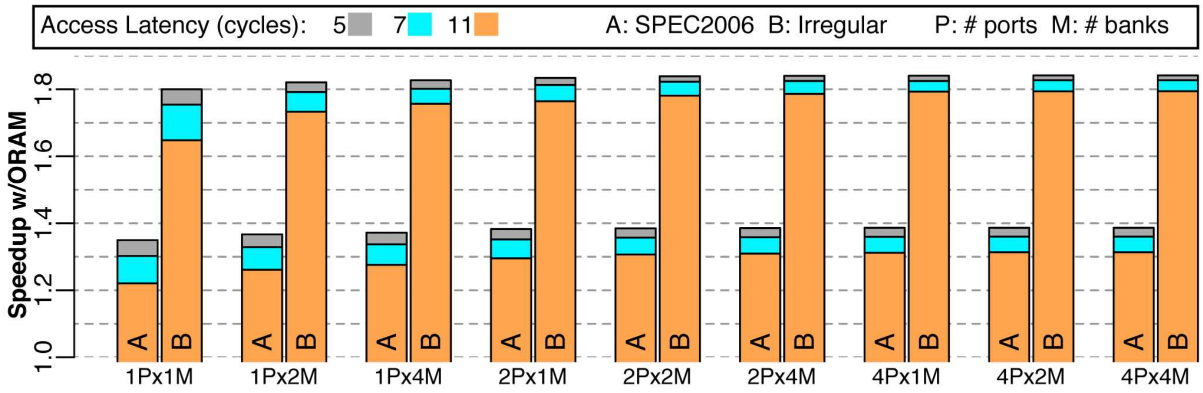


Figure 3.7: Overall Speedup of processor with a L1 cache and o-RAM. On each set, the first bar (A) represent SPEC2006 and the second one (B) irregular applications.

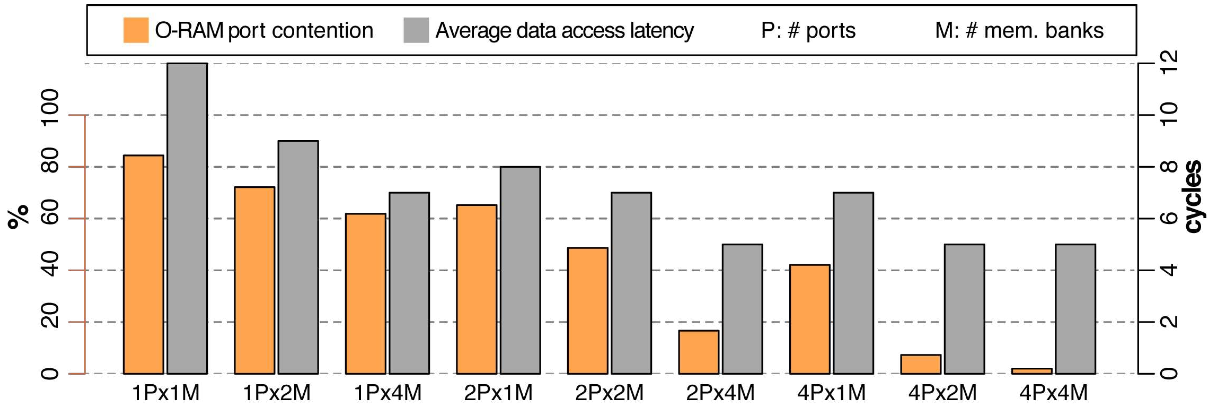


Figure 3.8: O-RAM port contention with SPEC2006 caused by access serialization on 7-cycle access latency

number of o-RAM banks or the number of ports. O-RAM system presents contention on its ports, as discussed in Section 3.1.1. Fig. 3.8 shows the percentage of times that a port P is busy when an operation is required, in the case of an o-RAM with 7 cycles access latency with SPEC2006. With (2P×2M) and (4P×1M) the contention is 48.6% and 42.1% respectively. Both obtained an average data latency of 7 cycles. Port contention has a direct effect on the data access latency. The 2P×2M and 4P×1M configurations have a good balance between area and performance, where both cases obtain an ≈ ×1.36 speedup.

From our results, we made two key observations. First, using o-RAM allows rethinking the cache hierarchy by reducing levels with similar latency to optical memory access and obtaining better performance to the baseline. Second, to obtain an average of ≈50% better performance, it is required ≈2 *ns* control and tuning for the optical switches. We conclude that further developments in optical devices could enable the full-optical memory architecture.

Scenario 2: Single core and Multicore with only L1 cache (reduced L1 Data Cache) and o-RAM,

Table 3.3 summarizes our setup configuration and details the platform cases similar to the evaluation in Section 3.3. In this new scenario, we perform two evaluations: a) a sensitivity analysis with a single core to explore the data cache size reduction, and b) a multicore evaluation.

Processor			
Frequency	2 GHz		
Cores	1, 4, 8		
Main Memory			
Type	DDR3-1066	o-RAM	
Latency (cycles)	150	5, 7, 11	
Cache			
	Baseline	Optical Single core	Optical Multicore
Levels	L1 I+D: 64 KB L2: 2 MB	L1 I: 64 KB L1 D: 32, 128, 512 B, 2, 8, 32 KB	L1 I: 64 KB L1 D: 2KB
Latency (cycles)	L1: 4	L1: 4	
Associativity	L1: 4-way L2: 16-way	L1: 4-way	
Block Size	32 B		
Policy	LRU		

Table 3.3: Configuration to evaluate the scenario with o-RAM and a single level cache with reduced data cache.

Single core with o-RAM. For this experiment, we used pinballs representation of important regions of code, enabling fast simulation. We used the SPEC06 benchmarks presented in Table 3.1 to explore the different configurations for o-RAM. Figure 3.9 presents the results for speedup comparing the electrical and the optical system. The results are grouped in columns by the size of L1D cache line size: 32 B, 128 B, 512 B, 2 KB, 8KB, and 32 KB; and each plot as a row shows a different estimation: speedup, average access memory latency and port contention on banks. In Fig 3.9 a) each L1D size configuration explores the numbers of ports (P) per o-RAM bank (B), where each combination shows three bars for the 5,7 and 11 latencies, respectively. The speedup reports the geometric mean of the execution of the applications of SPEC2006, as shown in Table 3.3; these results are normalized to the execution of the electrical system. If the values are higher than baseline 1, the result is exhibiting speedup; otherwise, it represents slowdown.

The speedup results show similar performance to the electrical system with a 2 KB L1D for the (2) ports per (2) banks with a latency of 7 cycles. With the 8 KB and 32 KB, the optical systems show an average speedup of 37%. In Figure 3.9b we identify that the average access latency for L1D 8K is 18 cycles. Figure 3.9 c shows that the speedup occurs when the percentage of port contention is below 30%. As expected, if the system has more ports or banks, the contention, and average memory latency drop down. However, this requires more resources that will directly impact the total area of the system. We also noticed that the results for 2 KB, 8 KB, and 32 KB are similar. The contention might limit the speedup of the single-core due to circuit switching.

Multicore with o-RAM. Using the results of the single-core as a start point, we explore the multicore system from Fig. 3.3 with $N = \{4, 8\}$ cores and L1D 2 KB with a latency scenario of 11 cycles. The applications executed are part of the PARSEC benchmark and are listed in Table 3.1. Figure 3.10 shows the multicore system results where each column groups the results per number of cores. Each bar represents a different number of ports (p) configuration per number of banks (b). The results show an average slowdown of 23% compared to the electrical system. The results show decreasing performance for the configurations with 1 and 2 banks (b) while increasing the number of ports for both 4 and 8 cores, which is different for the single core evaluation in the sense of more resources available allows better performance.

We conclude from the results of using a smaller L1D cache without L2 in the two scenarios. First, a single core with o-RAM can achieve similar performance to a conventional baseline system. Despite the sequential access required by setting paths, o-RAM low latency (tens of ns) produces short processor stalls (tens of cycles). Second, multicore with o-RAM performs worse than a multicore baseline system, caused by the increasing miss occurrence of the core requests over different cache lines. This is dependant on the workload and application type. In Chapter 5, we perform a thorough analysis of multiple benchmarks with multicore and memory systems with photonics in data centers.

3.4 Final Considerations

We proposed a full-optical memory for the main memory system. A full-optical memory benefits a computing system's performance because it could provide: a) constant operation latency and b) memory hierarchy reduction. We identified critical challenges related to optical memory and switch fabrication that could enable such a system. However, our exploration allows for understanding the current limitations of photonics for the main memory. With the selected benchmarks, we observe that a full-optical memory can lead to a 32× reduction of the L1 data cache and an average slowdown of 12%, and a maximum slowdown of 23% for multicore processors with the selected benchmarks. We conclude that such a full-optical RAM is far from feasible to fabricate because it is required to obtain a switch with ≈ns of operation latency, including control delay. As switching control is reduced and fabrication is mature, future works need to address the design of a control layer for the optical switches.

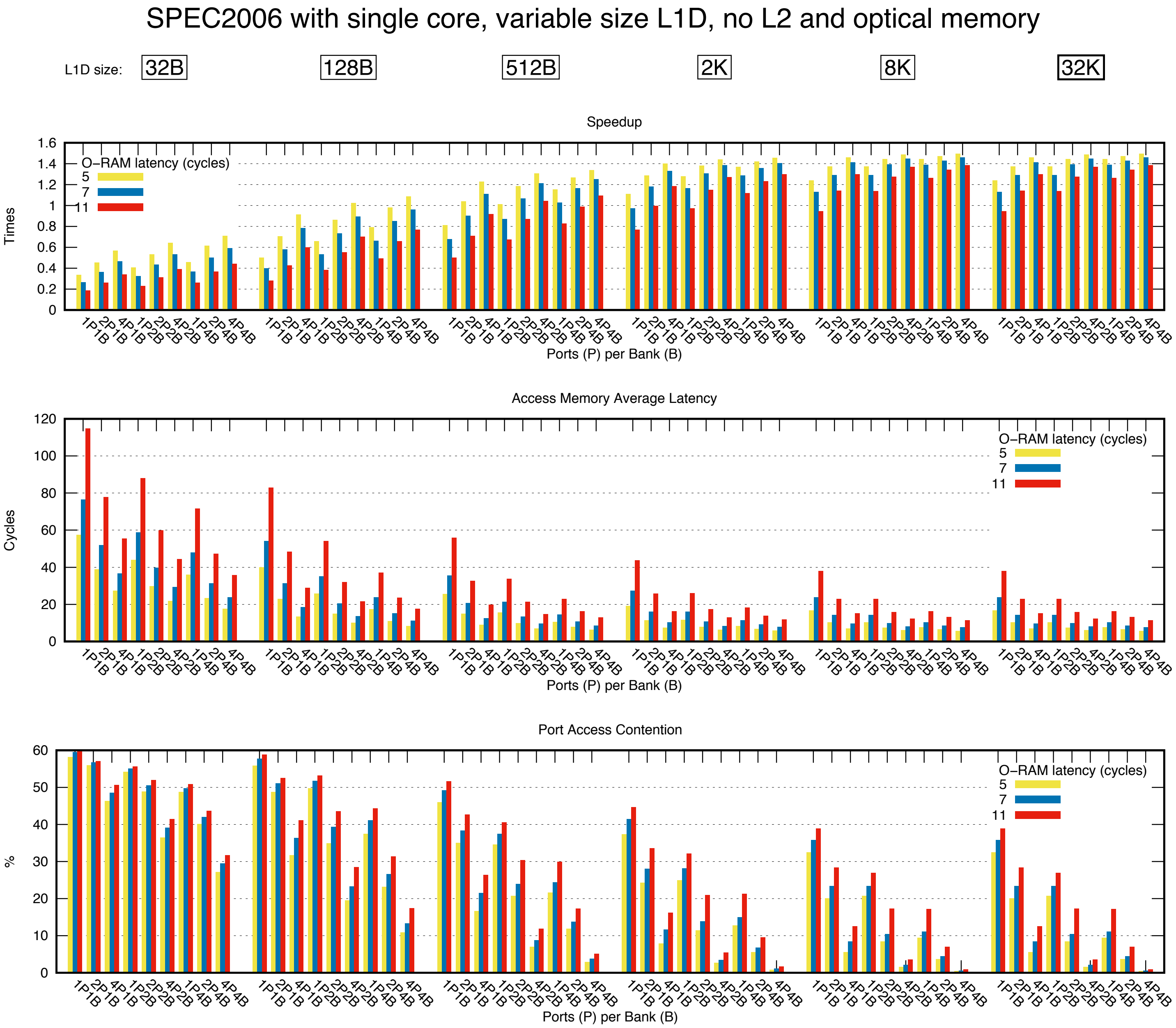


Figure 3.9: SPEC2006 results wih a single core with an optical RAM and a reduced L1D (DBuffer). a) Speedup b) Average Memory Latency c) Port contention

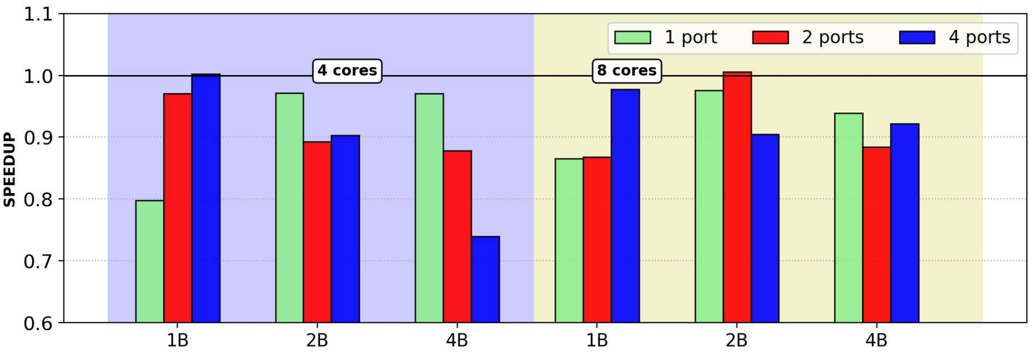


Figure 3.10: PARSEC speedup with an optical RAM and a reduced L1D (DBuffer)

Chapter 4

Reconfigurable Optical Multi-GPU Architecture

In this chapter, we propose the usage of optical interconnects for inter-GPU communication. We analyze the NVLink characteristics, NVIDIA interconnection for GPU point-to-point communication as a baseline to propose our reconfigurable optical multi-GPU architecture. Our approach is extendable to future and similar interconnections from other major vendors (e.g., AMD). In 2018 [15], we included our case study results with a set of well-known machine learning models, showing promising results. In the following sections, we 1) propose our optical multi-GPU architecture, 2) detail a bandwidth steering algorithm for reconfigurable architecture, and 3) show results with data rate improvement up to 20%.

4.1 Multi-GPU Computing

A Graphics Processing Unit (GPU) is a *processing element* composed of multiple cores, where each core execute *single-instruction multiple-thread* (SIMT) programs. Figure 4.1 shows the architecture of a GPU processor. A single GPU card has multiple processing cores on a single chip package, which is connected to high-bandwidth external DRAM chips (e.g., GDDR5 achieving 28 GBps, or HBM2 delivering >100 GBps). The GPU chip has an internal interconnection network to share data between the caching devices (*local memory*) and the off-chip DRAM memory. For example, the V100 GPU product from NVIDIA [189] has 80 SIMT cores and up to 32 GB of HBM memory. Each SIMT core has five types of units: 1) 64 *single-precision floating-point* (FP32), 2) 32 *double-precision floating-point* (FP64), 3) 64 *single-precision integer*, 4) 8 *tensor cores*[1], and 5) 4 texture units.

Multi-GPU computing is the preferred platform for the development of machine learning applications (e.g., deep neural networks and convolutional neural networks) in deployed data-centers, such as Facebook, where they process daily $\approx$ 3 billion images with convolutional networks [106, 261, 134]. The main reason to use GPU is its high performance executing matrix-matrix multiplications (also known as, *general matrix-matrix*

[1] A *Tensor* core is a dedicated unit for 4×4 matrices operations in floating-point precision.

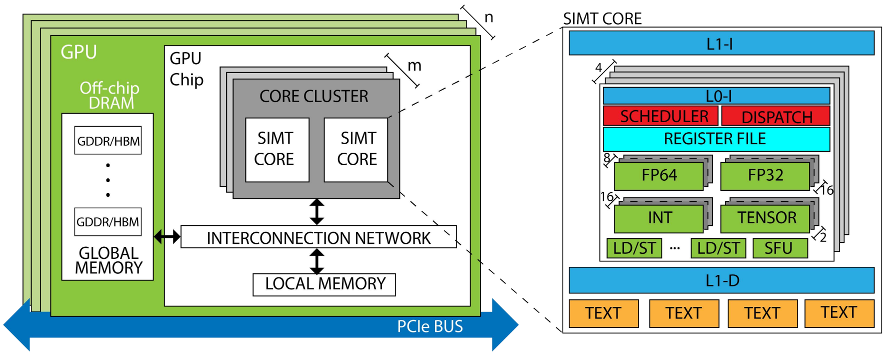


Figure 4.1: High-level architecture of a GPU

multiplications or GEMM) on its tensor cores. A tensor operation, which is the core of convolution operations in machine learning applications, consists of a matrix-multiply and accumulation defined as $D = A \times B + C$, where A and B are FP16 input matrices, C is an FP16 or FP32 matrix. Then D is an FP16 or FP32 result matrix depending on its inputs. There exist other types of dedicated hardware for machine learning, such as the Tensor Processing Units (TPU) [118], photonic tensor cores [176], and FPGA accelerators [120, 74]. Please refer to [242, 134], for a perspective on machine learning hardware.

4.2 Overview of Deep Neural Networks

Deep Neural Networks (DNNs) solve complex problems in diverse domains, such as image classification, speech recognition, and genome analysis. A DNN is a Machine Learning (ML) model, from the subset of Deep Learning (DL) in supervised learning, that emulates the process of human learning over time by estimating the adaptable parameters of the model (also known as weights). Figure 4.2 shows a DNN example of classification, where the model receives lung images as input, and the output is a medical diagnosis. The DNN model has three types of layers: 1) a single input layer, that receives raw data, 2) N hidden layers, formed by multiple features implemented through the learning process by discrimination of other irrelevant features, meaning that, after passing a layer, the output is more refined, and 3) an output layer, representing the result of the model. A layer is a set of multiple units called *perceptrons* that behave as a biological neuron.

A DNN model exhibit two identifiable phases during the learning process execution [135], depicted in the classification example of Figure 4.2a. First, the DNN uses a large data sample set in the *training* phase, to produce a highly accurate output adjusting the weight vectors on each layer. Training requires parallel and computing-intensive execution by forward passing (red arrows) the DNN model, estimating gradient vectors of the error for each hidden layer and, back-propagating (blue arrows) the gradients to adjust and update the weights to reduce the error. Second, the trained DNN model is fed with new input data in the *inference* phase for classification; this process is equivalent to the forward passing (blue arrows) operations and does not require back-propagation.

Figure 4.2b shows a perceptron, based on the original model proposed by Rosenblatt in [215], which has a set of inputs $x_1, x_2, \vdots, x_n$ that are important *features* for the object (i.e., image) classification. For each feature, there is a correspondent weight representing the degree of importance $w_1, w_2, \vdots w_n$. We could define the learning process y_1 as an activation function of the weighted sum of all the features. The step, sigmoid, and sign functions are commonly used for activation; for example, in case of a step function ϕ the output of y_1 is -1 if $\sum_{i=1}^{n} w_i xi$ is $\leq$ threshold, or 1 if it is $>$ threshold.

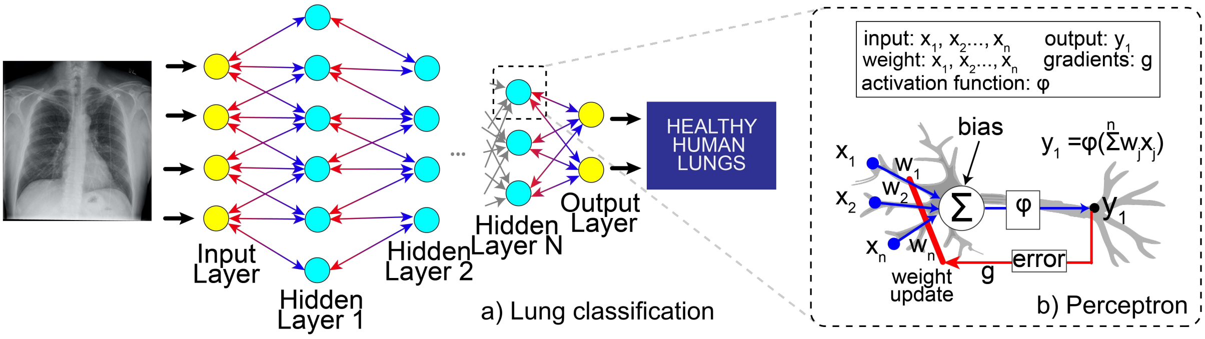


Figure 4.2: DNN model

4.2.1 DNNs evolution with multi-GPUs

As researchers adopted GPUs as the main device for applications that requires an increasing number of threads with massive task execution in parallel (e.g., deep learning [108, 106]), current high-performance data center need to support multiple GPUs on single nodes and scale them by clustering several of those nodes. Figure 4.3 shows a timeline that relates popular machine learning works of the decade with multi-GPU interconnection support. Please refer to the comprehensive survey in machine learning [12], where the authors provided detailed information about the network models.

We identified three recent periods on GPU hardware after the release of the programming model CUDA [127], and the seminal work [50] in parallelization of a convolutional network model on GPU.

Those periods also occur after the public release of the ImageNet database for machine learning classification [70], which contains a sizeable visual image data set for training and is used in the annual ImageNet Large Scale Visual Recognition Challenge (ILSVRC).

1. During the **first period**, GPU vendors enable direct peer-to-peer (P2P) access, equivalent to Remote Direct Memory Access (RDMA), between different GPUs. Initially, NVIDIA releases GPUDirect, and later AMD announces similar features for their HyperTransport products. During this epoch, in 2011 IBM Watson won a Jeopardy contest against human opponents that allowed deep learning to gain the spotlight from international media [84, 164]. In [131], the authors proposed the Alexnet model in a multi-GPU implementation, being the first time a deep convolutional network won the ILSVRC 2012, obtaining the lowest error rate of $\approx$ 15.3%.

2. The **second period** characterized by network RDMA operations allowed in GPU device memory (NVIDIA [184] and AMD GDMA [14]), allowing communication

.

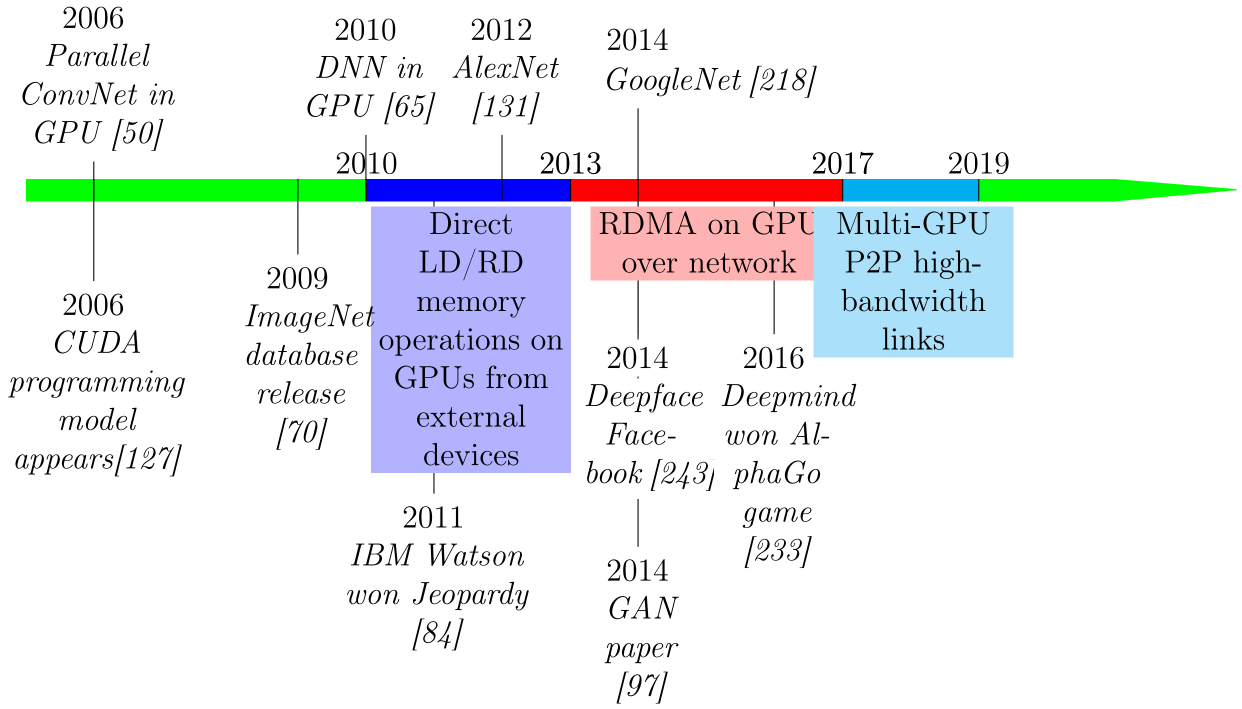

2006
Parallel ConvNet in GPU [50]
2010
DNN in GPU [65]
2012
AlexNet [131]
2014
GoogleNet [218]
2010
2013
2017
2019
Direct LD/RD memory operations on GPUs from external devices
RDMA on GPU over network
Multi-GPU P2P high-bandwidth links
2006
CUDA programming model appears[127]
2009
ImageNet database release [70]
2011
IBM Watson won Jeopardy [84]
2014
Deepface Face-book [243]
2014
GAN paper [97]
2016
Deepmind won Al-phaGo game [233]

Figure 4.3: Timeline

between GPU memories located in remote devices. During this period, the deep learning models consistently won the ImageNet contests, as the GoogLeNet model [218]. Other models emerged as the ResNets [244], and the Generative Adversarial Network (GAN) [97], which is an unsupervised model. Two events mark this timeline, Facebook unveils details about their face recognition program, Deepface [243], and Google's Deepmind program defeat the human champion of the Go game using 176 GPUs [233].

3. Today, we are facing the **third period**, with multi-GPU platforms using high-bandwidth interconnects between the processing devices. NVIDIA introduced their P2P link for GPUs called NVLINK [189, 87], which allows fast transactions without using the traditional Peripheral Component Interconnect Express Bus (PCIe). This period is mainly focused on models with high connectivity between layers to reduce network parameters, such as Densenet based models [111].

While larger (width of layers) and deeper (number of layers) models are developed using larger training sets to obtain more accurate results [12], multi-GPU services need to move towards Tbps communications for the processing elements, providing a higher abstraction level in the programming model to alleviate the effort of the DNN designer.

4.2.2 Parallel DNNs execution in multi-GPU

Figure 4.4 shows the DNN execution in a multi-GPU platform using the *parameter server* approach. In [144, 143], authors proposed the parameter server for scaling distributed machine learning workload. A parameter server is a technique used in distributed machine learning. Its objective is an efficient parameter update and synchronization (i.e., weight and gradient) across the multi-GPU. Multiple servers share the parameters globally and reduce a bottleneck by distributing access over the servers, where each server maintains a portion of data. A group of GPUs, named workers, can be associated with a specific server.

In Figure 4.4, the platform has two GPUs operating as workers, a single CPU as a task manager for the workers, and two external parameter servers. We assume that the DNN layers execution is distributed among the workers in five stages:

1. At the beginning of the execution, the CPU receives variables from parameter servers, such as training data.
2. The CPU transfers the variables to the workers according to each one scheduled execution.
3. With the received data, the GPUs estimate the gradients (forward) iteratively.
4. A backward transference from the GPUs to the CPU occurs with the gradient results.
5. Using the obtained gradients, the CPU aggregates them into a unified gradient vector.
6. Finally, the CPU updates the parameter servers.

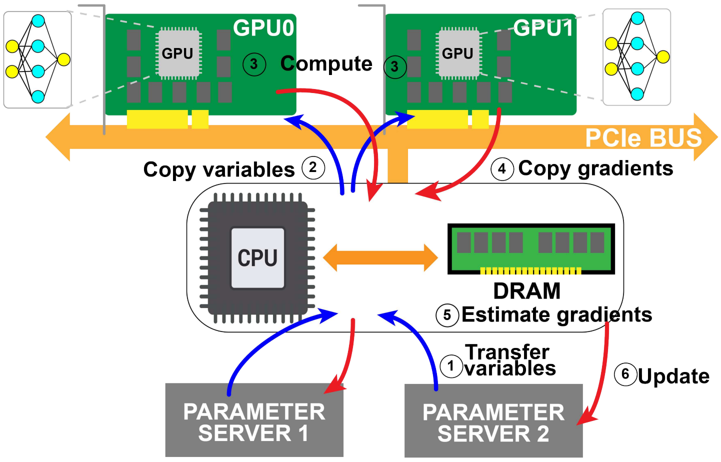


Figure 4.4: DNN model execution on a multi-GPU platform

DNN execution on a GPU uses the tensor cores units to achieve high throughput in the TFLOPs order. However, bandwidth limits the overall performance [104]: 1) the memory bandwidth that involves layer activation and data access at tensor cores operation level, and 2) the device interconnection bandwidth that constraints the gradient exchange. As discussed in [104], there exist several compression techniques for DNNs to optimize bandwidth utilization; the most studied approaches are based on quantization, sparsification, pruning, decomposition, and distillation [33, 228].

In this work, we focus on training execution because of two key reasons. First, it is more computationally demanding than inference because of the large size of the input set [261]. Second, when backward propagation occurs, then training requires more communication between the execution nodes than inference [22]. In [147], the authors proposed Deep Gradient Compression (DGC) to reduce the interconnection usage in the distributed scenario for training.

Also, DNN training and inference performance can speed up using new hardware accelerators. Table 4.1 shows recent DNN dedicated hardware in six different domains. Notice that optical accelerators appeared with a potential execution time of matrix multiplication in constant time $O(1)$ operation [115]. For more information about DNN accelerators, please refer to the following surveys: 1) FPGA accelerators [100], 2) optical accelerators [68], and 3) implementation challenges [211, 39].

Table 4.1: DNN hardware accelerators

Category	Work
Systolic	[205, 89]
Emerging memory	[18, 155, 225]
Optical	[19, 231, 115, 176]
Processing element clustering/Network-on-Chip	[133, 53]
Approximated	[17, 137]
Cloud	[88]

4.3 Multi-GPU Interconnects

There are two main factors for multi-GPU platform seamless implementation in data centers. First, a unified memory programming that helps the developer and supports both: a) GPU-to-host (or GPU-to-CPU), b) GPU-to-GPU (or GPU peer-to-peer) communications. Second, a high-bandwidth and energy-efficient interconnection between GPUs. Both conditions could lead to efficient implementations of machine learning models, that necessitate efficient data movement for gradient propagation.

NVIDIA's programming model introduced the Unified Virtual Addressing (UVA) and later the Unified Memory (UM) manager [184, 187], for efficient memory transactions by sharing data while executing a single application in multi-GPUs. CUDA API supports two types of memory management, and the developer could use both approaches in parallel programming:

1. **With explicit memory functions:** Figure 4.5 shows the operation of `cudaMalloc()` and `cudaMemcpy()` functions highlighting their differences. With `cudaMalloc()`

(red), the CPU allocates (or pins) a segment on its DRAM modules and shares it with the GPU, then the GPU caches can modify data in the host memory accessing trough the CPU without having a local copy. The `cudaMemcpy()` (blue) function allows direct memory transactions bypassing the host, and directly reaching the target trough the interconnect. It is allowed to perform direct operations between GPU-GPU, CPU-GPU, storage-to GPU (e.g., `cudaMemcpy() depicted in black lines to an non-volatile memory device express (NVMe)`), and GPU with a third party devices using a NIC.

2. **Using an automatic page migration:** CUDA extended their original explicit approach, and implemented the allocation function `cudaMallocManaged()` to declare a global pointer accessible from any device (CPU or GPU). Different than the first approach, there is an automated page manager based on heuristics to perform allocation on device prefetch or its first access, flexible page migration, and reduce the page faults. However, similar to the zero-copy accesses with `cudaMalloc()`, it is also possible to use `cudaMemAdvise()` to give hints to the page manager to pin pages, and use `cudamemPrefetchAsync()` for explicit page migration.

Unified Memory (UM) research is focused at overcoming memory management problems, such as complex asynchronous programming and redundant page migration overhead [276], handling memory oversubscription with prefetching for efficient page migration due to *far-faults* [91], and speedup performance in multi-GPU execution with MPI [63, 64, 214]. From the co-design perspective, hardware mechanisms can help improve data migration efficiency further, which motivates us to study the multi-GPU interconnection.

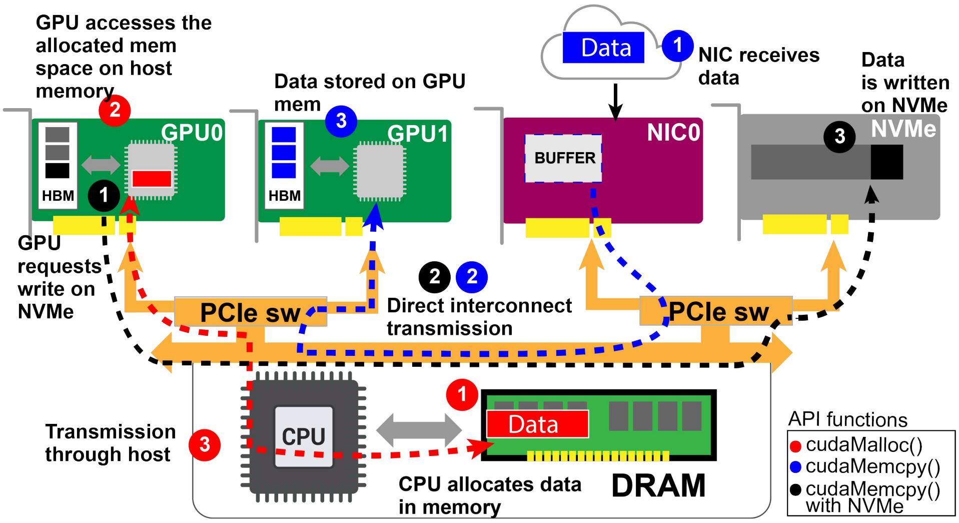


Figure 4.5: DNN model execution on a multi-GPU platform

4.3.1 High-level architecture of multi-GPU interconnects

Efficient transactions between processing elements have a critical role in achieving high-performance computing (HPC) in data centers. Expected HPC performance varies according to the application domains and workloads (e.g., web services [47], deep learning

[62]). Performance is also related to the characteristics and constraints in the data center, such as heterogeneity of the processing elements, interconnection bandwidth, node topology, and maintenance cost. As more massive clusters are required to solve complex problems, the interconnect technology directly impacts three main aspects [227]: i) execution scaling (inter- and intra-node), ii) inter-job interference, and iii) communication patterns. Table 4.2 shows the actual top-five supercomputers according to `TOP500.org` [234]. We observe two interconnection characteristics: 1) all of the data centers adopted fat-tree topology for intra-node communication, 2) NVIDIA's proprietary NVLink is the preferred interconnect for inter-node communication of multi-GPUs. Intra-node interconnect converges in the use of links with up to 200 Gbps and very well known topologies such as Fat Tree [114] or Dragonfly [48]. However, inter-node communication for multi-GPU is still an active field of study since the introduction of NVlink to complement current PCIe usage. We outline the main architectural characteristics of multi-GPU interconnects.

Table 4.2: First five Top500 - June 2020

Ranking	Name	Location	Nodes	Config/Node	Interconnect	Topology
1	Fugaku	JAPAN	158976	48 Fujitsu A64FX 4 Fujitsu A64FX	Fujitsu Tofu-D [5]	Torus
1	Summit	USA	4096	2 IBM Power9 6 NVIDIA V100	Infiniband EDR [1]	Fat tree
2	Sierra	USA	4320	2 IBM Power9 4 NVIDIA V100	Infiniband EDR [1]	Fat tree
3	Sumway	CHINA	40960	1 SW260101	Sumway	Fat tree
4	Tianhe	CHINA	16000	2 Intel Ivy Bridge Xeon 3 Xeon Phi	TH Express2 [2]	Fat tree

[1] up to 100 Gbps [168]
[2] approx. ≈ 100 Gbps [114]
[3] up to 200 Gbps [168]
[4] separate subsystem with NVIDIA GPUs [167]
[5] up to 100 Gbps [72]

Peripheral Component Interconnect Express Bus (PCIe): PCIe is a high-speed serial bus used for CPU communication with discrete devices such as NVMe storage, network interfaces, and accelerators. Access via PCIe is in the order of μs, and it is slower than the main memory access, which is in the order of ns. Figure 4.6a) shows the topology for a state-of-the-art PCIe based multi-GPU system with a maximum of 16 GPU nodes (i.e., DGX2 servers by NVIDIA [186]). Multi-GPUs with PCIe interconnects have a complete binary tree topology, where the CPU processor is the root node, and GPUs are the leaves; then, the PCIe switches serving as virtual PCI bridges enable the branches. For such an extensive multi-GPU system with 16 GPUs, two separate trees with 8 GPUs each are implemented with height equal 3, and required a multilevel (two levels) PCIe switch network to communicate them. CPUs can communicate with each other using a dedicated P2P interconnect (i.e., Intel Quick Path Interconnect [277]). Figure 4.6b) depicts a PCIe bidirectional link, which is formed by $N = 1, 4, 8, 16$ lanes, and each lane has a set of wires for interconnection. A PCIe 3.0 link, which is used by current GPUs, is bidirectional and uses up to ×16 lanes, four (unidirectional) wires form each lane working as a differential pair delivering 2.5-16 Gbps. The maximum bandwidth of the PCIe 3.0 link is 15.75 GBps, with a total of 164 pins, being 64 pins for the differential pairs.

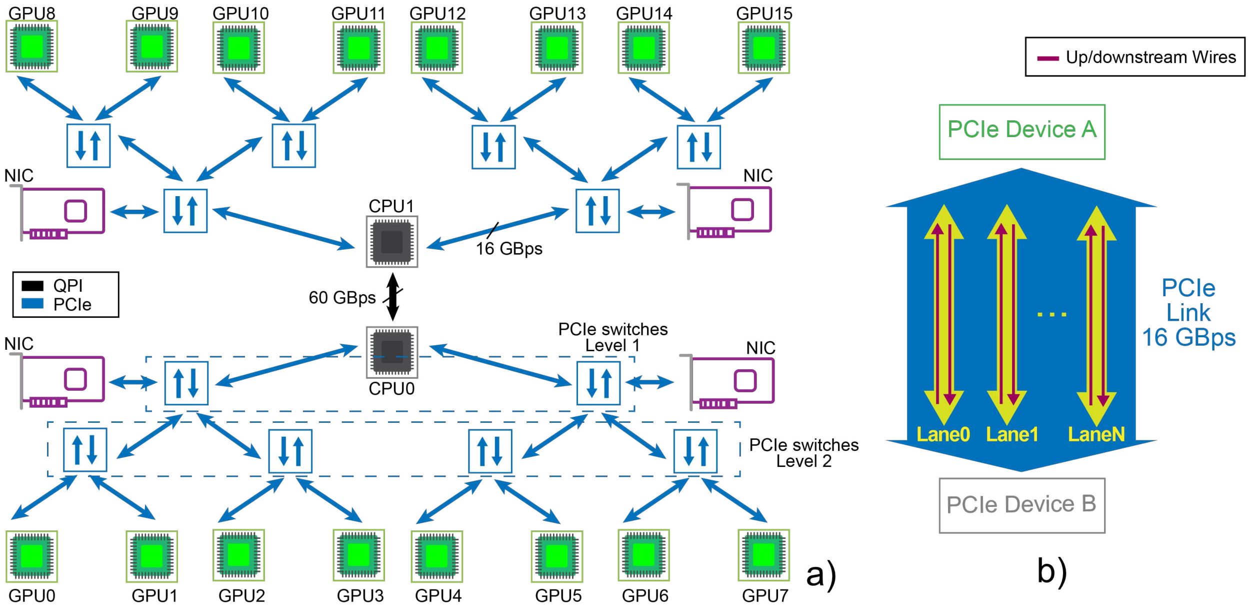


Figure 4.6: Multi-GPU with PCIe interconnection. a) Tree topology with 6 PCIe switches for sustaining 8 GPUs per CPU. b) 16 GBps PCIe link formed with N lanes, where each lane has a pair of wires.

NVlink. NVLink is a P2P communication interface developed by NVIDIA [87], and it is envisioned to support GPUdirect operations: i) providing higher bandwidth than traditional interconnects such as PCIe, and ii) increasing the interconnectivity support for a higher number of devices. Figure 4.7a) shows the topology for a commercial server using NVLinks (i.e., NVIDIA DGX1 [185]). In this system, there are 2 CPUs with a total of 8 GPUs (GPU0 to GPU7) distributed evenly to each one. For example, CPU0 can access GPU0 to GPU3 using a PCIe interconnect with a tree topology. Therefore, in this node arquitecture, there are two PCIe switches per CPU and two GPUs per PCIe switch. GPUs have 50 GBps NVLink interconnects for direct access between them, with a maximum of 6 NVLinks per GPU. NVLinks interconnections define a hypercube mesh topology configuration between the GPUs. Figure 4.7b) shows the implementation of a 50 GBps NVLink. An NVLink is a bidirectional link composed with two unidirectional sublinks, where each sublink has 8 unidirectional NVIDIA High-Speed Signaling Interconnects (NVHS) [56]. NVLink operation is packetized using messages from 16 Bytes to 128 Bytes, transmitted in smaller flit units of 128 bits.

NVSwitch. In [188], NVIDIA presented its NVSwitch product for the DGX2 server. NVSwitches form a fully connected crossbar for non-blocking multi-GPU (total of 16 GPUs) interconnection in DGX2. The crossbar extends the PCIe network from Figure 4.6a) to allow direct GPU communication. Figure 4.8, shows the crossbar architecture and details the NVSwitches. There are 12 NVSwitches, distributed evenly in two baseboards (6 per board), where each baseboard connects 8 GPUs. A GPU is connected to a baseboard

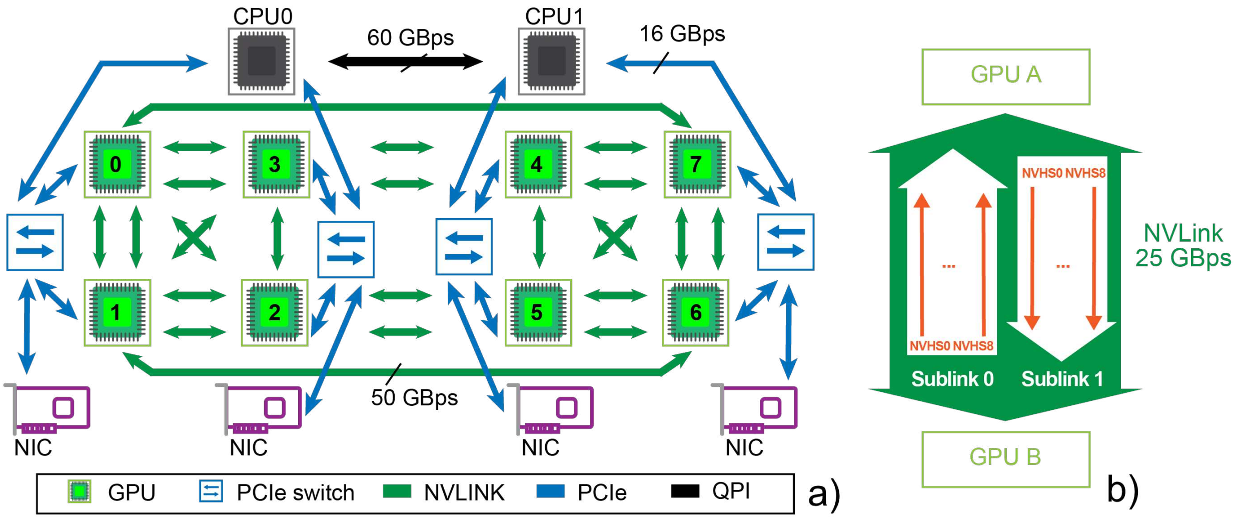


Figure 4.7: Multi-GPU with NVIDIA NVLink interconnection. a) Hypercube topology with 8 GPUs and 4 PCIe switches, 2 per cube side. b) 50 GBps NVLink with two 25 GBps unidirectional links.

using 6 NVLinks, one to each NVSwitch. There is a connection between baseboards using 8 ports per NVSwitch. Notice that a single NVswitch has 18 ports, and in DGX2 only 16 are used, allowing future interconnection increment. The aggregated throughput in one crossbar hop is 300 GBps, considering the 50 GBps × 6 NVLinks. The aggregated bandwidth for baseboard communication is 2.4 TBps estimated as 50 × 8 used NVLinks × 6 NVSwitches.

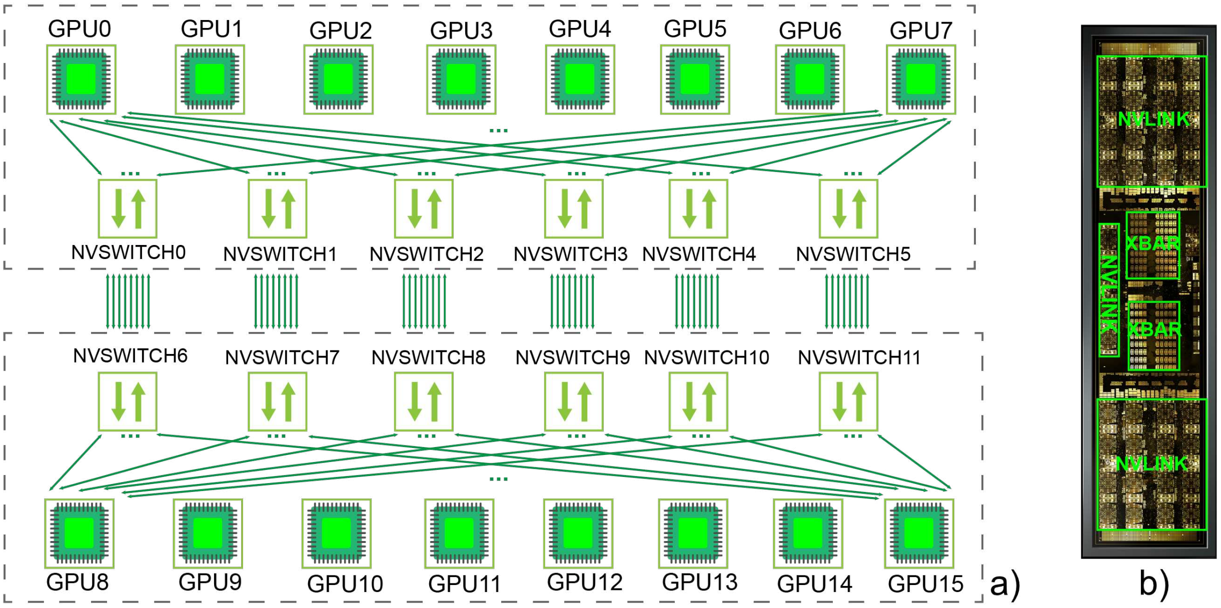


Figure 4.8: Multi-GPU with NVIDIA NVSwitch interconnection. a) All-to-All topology with 16 GPUs, 8 per side. b) NVSwitch chip photography from [188].

In [139], the authors evaluated multi-GPU interconnects observing that NVLink and PCIe produce high non-uniform memory access (NUMA effect) during execution, while NVSwitch memory access is uniform (UMA). The authors also highlighted the importance of developing programming models to support diverse interconnect architectures and to improve performance modeling to evaluate different scheduling techniques and data migration scenarios in multi-GPU.

4.4 Optical reconfigurable architecture for multi-GPU

NVLink and NVSwitch alleviates the bandwidth-related problems on systems that only use conventional PCIe interconnects and improves the performance in multi-GPU platforms. However, such interconnects face three main scalability issues due to electrical constraints [9, 227]. First, increasing the number of parallel wires required to achieve higher bandwidth brings new crosstalk challenges. Second, the interconnect distance is limited to onboard distances in the order of mm, not being compatible with rack aggregation using the interconnect. Third, electrical switches ports bring new scalability challenges as a higher number of GPUs require peer-to-peer communications. In this line, we study Silicon Photonics (SiP) to design an efficient and high-bandwidth multi-GPU architecture that alleviates those problems [173], and specifically, we use an optical switch as the enabling device for such architecture [61].

SiP optical circuit switches (OCS) allow dynamic link-level reconfigurability for a new class of bandwidth-steered applications. In [257, 232], the authors showed that bandwidth-steering via OCS could achieve significant performance gains for dragonfly networks.

We propose an optically connected GPU architecture using SiP circuit switches, as a new GPU communication structure to increase performance further. We introduce an exhaustive bandwidth-steering algorithm to minimize total latency for device memory transfers. Initial results for this algorithm and proposed architecture are reported in Section 4.5.1 using traffic traces generated from DNNs execution on a real multi-GPU system.

Figure 4.9a shows our optical multi-GPU architecture that enables high-bandwidth and efficient peer-to-peer communication based on reconfiguration. In our architecture, an Optical Circuit Switch (OCS) connects all the GPUs as a central arbiter of the SiP links. Each SiP link uses Wavelength Division Multiplexing (WDM), achieving high bandwidth via multiple *virtual* paths (please refer to Chapter 2). In our architecture, additional logic monitors the traffic between GPUs and controls the OCS. We use bandwidth steering that reassigns the optical channels (i.e., wavelengths) with low data rates to GPU-to-GPU links with an increasing or more demanding data rate. Using reconfigurable OCS brings to the multi-GPU four key features: 1) avoidance of link overprovision, 2) performance improvement by reducing the transaction time due to bandwidth *reallocation*, 3) delivers the flexibility to create adaptable topologies, and 4) enables GPU scalability because of the continuous improvement in photonics achieving higher bandwidth than electrical interconnects.

Figure 4.9b shows the implementation of a 50 GBps Silicon Photonics (SiP) link, formed by two 25 GBps links. Every 25 GBps optical sublink could be implemented with 8 × 25 Gbps optical links considering that state-of-the-art photonics reaches up to 1 Tbps bandwidth per link [24, 26]. Notice that as photonics technology progress, it is expected

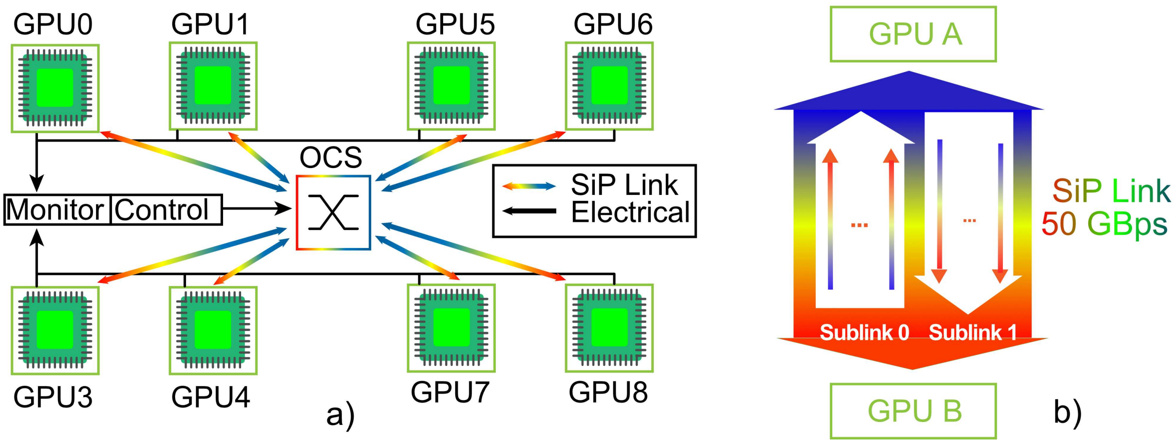


Figure 4.9: Optically Reconfigurable Multi-GPU architecture. a) Hypercube topology with 8 GPUs and an optical circuit switch (OCS) that enables reconfigurability. Notice that being reconfigurable, allows to change the topology according to the workload requirements. b) 50 GBps Silicon Photonics (SiP) link based on two 25 GBps unidirectional links.

that monolithic fabrication could be feasible [94, 230].

Figure 4.10 shows a simple example of the bandwidth-steering concept to optimize the outgoing connections on a 3 GPU system (GPU A, GPU B, GPU C). Depending on the microarchitecture of the OCS, the optical signals could each represent: i) a single wavelength signal, or ii) a wavelength-division multiplexed (WDM) signal. The first case scales worse than the second one because each wavelength requires an optical fiber, therefore increases the number of ports on the OCS. In [59, 112], the authors demonstrated scalability in a broadband WDM optical switch using microrings. Despite there still exist challenges related to the number of ports, wavelengths, and losses; those works show a compelling step in optical switching for high-bandwidth computing systems. Considering that the switching operation is in the order of μs and HPC applications (e.g., deep learning) produces a long execution time on multi-GPUs, our architecture can dynamically reconfigure the OCS to provide larger portions of the total bandwidth to the highest-traffic GPU pairs (e.g., GPUA to GPUB): 1) at the beginning of each application phase, and 2) within a phase. Fig. 4.10b illustrates a situation in which the algorithm has determined that an unequal amount of bandwidth between GPU A and GPUs B and C will result in increased performance. The specifics of this configuration algorithm are detailed in the following Section 4.4.1.

4.4.1 Bandwidth Steering

Several works [245], evaluate OCS for HPC system communications to optimize the large data flows between computing nodes (rack or servers). In [257] the optical Flexfly architecture for HPC system was proposed that allows reassignment of under-utilized links to a group of computing nodes with saturated communication. Such an approach is known as bandwidth steering and can be used for multi-GPU communication. Notice that bandwidth steering can be used only in systems with irregular traffic patterns [232]; thus,

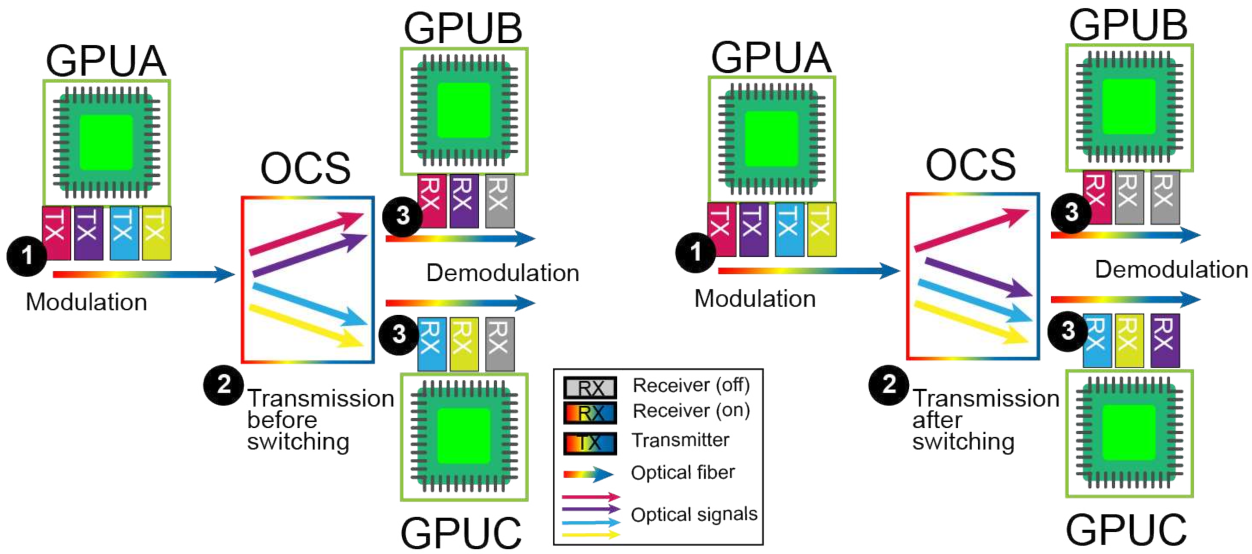


Figure 4.10: Based on the relative amounts of data sent from GPU A to GPUs B and C during a given application, the optical circuit switch (OCS) will reconfigure itself to minimize GPU A's total transmission latency. (a) The baseline configuration provides equal bandwidth from GPU A to GPUs B and C (b) In this example, the optimization algorithm has determined that GPU A requires 3x more bandwidth to GPU C than GPU B.

the traffic flow is not evenly distributed among the links of the network topology. However, for the state-of-the-art multi-GPU platforms (such as NVLink) we identify two main differences:

1. The complexity of the bandwidth-steering approach from [257] can be reduced to a minimization problem because the ideal communication is peer-to-peer, where the GPUs do not support routing of any kind, differently than HPC systems or data centers where the computing nodes have switches on top-of-rack (ToR).

2. GPU operation is in ns order, and the OCS induced latency can lead to performance degradation, while network communication latency is in μs the OCS latency can be hidden in between data transmission. This limitation motivates the analysis of traffic in multi-GPU platforms to enable dynamic optical reconfiguration.

Algorithm 1 shows the bandwidth steering minimization in pseudo-algorithmic representation. The bandwidth steering algorithm uses as inputs the traffic matrix generated during application execution and the connectivity matrix that represents the multi-GPU initial topology. All the GPU-to-GPU communication must have a non-zero amount of bandwidth allocated. Therefore to connect two GPUs, there must be a group of sublinks assigned. After reserving the minimum amount of allocatable bandwidth, referred to as a *bandwidth unit*, for every required connection, the remaining bandwidth units are distributed among the other GPUs to minimize the total amount of relative transmission latency (RTL). To calculate the RTL for a single source-destination pair requires the division of the element `[i][j]` in the traffic matrix for a given application by the corresponding element in the topology's connectivity matrix, where `[i]` and `[j]` represent the destination and source GPU IDs respectively. An aggregate RTL for each GPU is calculated by summing the RTLs from the appropriate column in the traffic and connectivity matrices. An exhaustive implementation is used to minimize the aggregate RTL for each GPU iteratively. This consists the calculation of $c_j * \binom{k-1}{k-c_j}$ RTLs per GPU where c_j is the number of required connections for GPU j, and k is the number of bandwidth units per GPU.

Algorithm 1 *MULTI–GPU–BW–STEERING(Traffic, Connect, numgpu)*

```
$RTL_{initial} \leftarrow Estimate\_RTL(Connect, Traffic)$
for $i = 1 \rightarrow numgpu$ do
  $Connect_{opt} \leftarrow feasible\_combinations(Connect[i])$ {Array of all possible combinations of redistributed links within a connectivity matrix column};
  for $j = 1 \rightarrow length(Connect_{opt})$ do
    $RTL = Estimate\_RTL(Connect_{opt}[j], Traffic)$
    if $RTL[j] < RTL_{initial}$ then
      $Traffic_{new} \leftarrow RTL[j]$
    end if
  end for
end for
```

4.5 Evaluation

We evaluated our optical reconfigurable multi-GPU architecture with bandwidth steering using an static post-mortem approach. We executed a set of machine learning applications and collected the traffic generated on an Amazon Server P3.16xlarge instance, which has a DGX-1 with NVLink and 8 V100 GPUs.

Figure 4.11 summarizes our evaluation methodology in four steps:

1. First, we define a connectivity matrix based on the topology. Fig. 4.9 shows our proposed reconfigurable architecture for 8 GPUs within a single server. The topology is based on the NVLink-connected hyper-mesh GPU configuration. The connectivity matrix for this topology is shown in the center. Each element `[i][j]` in the connectivity matrix denotes the total number of bandwidth units that GPU `j` can use to send data to GPU `i`. In this experiment, each bandwidth unit represents 1 NVLink sublink that provides 25 GBps of bandwidth. Each NVLink-enabled GPU has 6 outgoing and incoming sublinks, giving a total of 300 GBps of bidirectional bandwidth [87]. It is important to notice that while Fig. 4.10 shows the outgoing and incoming connections as a single bidirectional link; our optimization algorithm treats each unidirectional connection independently.

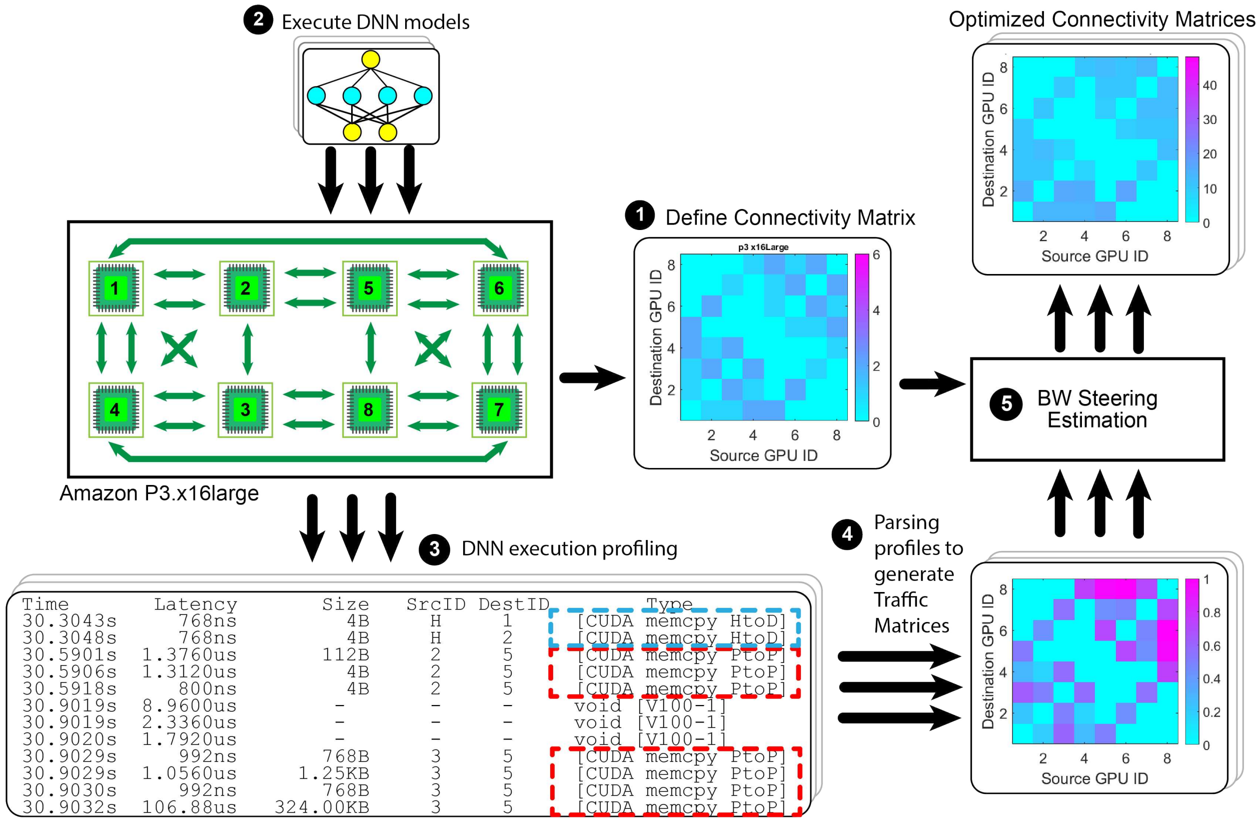
2. Execute the DNN models on the server with multi-GPU.

3. We captured traces from the training phases execution of deep learning models generated using Tensorflow [3]. Tensorflow automatically allocates tasks on the GPUs based on the infrastructure. The models used the parameter server stragegy for

gradient propagation (see Section 4.2, and we also evaluated variable update with replication where the CPU immediatly update the values on both parameter server and the local GPU copy after the gradient aggregation. The traffic capture was performed using the NVprof tool that creates an output log for 5 types of operations: a) `memcopyHtoD`, that represents host memory to GPU memory operation, b) `memcpyDtoH`, which are all the operations from GPU memory to host memory, c) `memcpyDtoD`, which are all GPU memory to GPU memory operations on the same devices, d) `memcpyPtoP`, that represents GPU memory to peer GPU memory operations (different GPU devices), and e) void, which are all the kernel execution operations.

4. We parsed the multi-GPU transaction logs to count the latency and data movements caused by `memcpyPtoP`, and generate traffic matrices.

5. We analyzed the reconfiguration mechanism with our implementation of the bandwidth steering algorithm, obtaining optimized connectivity matrices.



```
Time          Latency        Size    SrcID DestID         Type
30.3043s        768ns          4B       H      1   [CUDA memcpy HtoD]
30.3048s        768ns          4B       H      2   [CUDA memcpy HtoD]
30.5901s   1.3760us          112B       2      5   [CUDA memcpy PtoP]
30.5906s   1.3120us            4B       2      5   [CUDA memcpy PtoP]
30.5918s        800ns          4B       2      5   [CUDA memcpy PtoP]
30.9019s   8.9600us             -       -      -   void [V100-1]
30.9019s   2.3360us             -       -      -   void [V100-1]
30.9020s   1.7920us             -       -      -   void [V100-1]
30.9029s        992ns        768B       3      5   [CUDA memcpy PtoP]
30.9029s   1.0560us        1.25KB       3      5   [CUDA memcpy PtoP]
30.9030s        992ns        768B       3      5   [CUDA memcpy PtoP]
30.9032s   106.88us      324.00KB       3      5   [CUDA memcpy PtoP]
```

Figure 4.11: Bandwidth steering methodology using the measured traffic from multiple deep learning model training.

We used Convolutional Neural Networks (CNNs) in our evaluation. The Cifar10 dataset [130] was used to train the AlexNet, DenseNet100, DenseNet40, NASNet, ResNet110, and ResNet20 models. The Flowers dataset [1] was also used, and provided input to both MobileNet and VGG16 models.

4.5.1 Experimental Results

Fig. 4.9a shows our proposed reconfigurable architecture for 8 GPUs within a single server. The topology is based on the NVLink-connected hyper-mesh GPU configuration. The connectivity matrix for this topology is shown in Fig. 4.11❶. Each element [i][j] in the connectivity matrix denotes the total number of bandwidth units that GPU j can use to send data to GPU i. In this experiment, each bandwidth unit represents 1 NVLink sublink that provides 25 GBps of bandwidth. Each NVLink-enabled GPU has 6 outgoing and incoming sublinks, giving a total of 300 GBps of bidirectional bandwidth [185, 186]. It is important to note that while Fig. 4.11❶ shows the outgoing and incoming connections as a single bidirectional link, the optimization algorithm (Fig. 4.11❺) treats each unidirectional connection independently.

Figures 4.12 to 4.15 show the results in a two-column format, where the first column represents the traffic matrices and the second column, the optimized connectivity matrix. In our evaluation, a bandwidth unit represents 1 NVLink sublink with 25 GBps of bandwidth. Each NVLink-enabled GPU has 6 outgoing and incoming sublinks, giving a total of 300 GBps of bidirectional bandwidth. Notice that the optimization algorithm evaluates each unidirectional connection independently.

The scale in the first column is the total data flow (in percentage), while in the second column is the number of 25 GBps sublinks. Our results are organized as follows:

- **Case A:** Figure 4.12 shows the results obtained with Cifar10 input and parameter server without replication. *Alexnet* shows improvement by balancing traffic across all nodes on the optimized connectivity matrix, considering it exhibits high traffic interchange with node 5. *Resnet* exhibits high traffic across all nodes, and its optimized connectivity matrix maintains this distribution.

- **Case B:** Figure 4.13 shows the results obtained with Cifar10 input and parameter server with replication. *Alexnet* and *densenet* models show improved connectivity. Both models have high traffic on node 5, and the resultant optimized matrix show improved connection balancing traffic with nodes 3, 6, and 7.

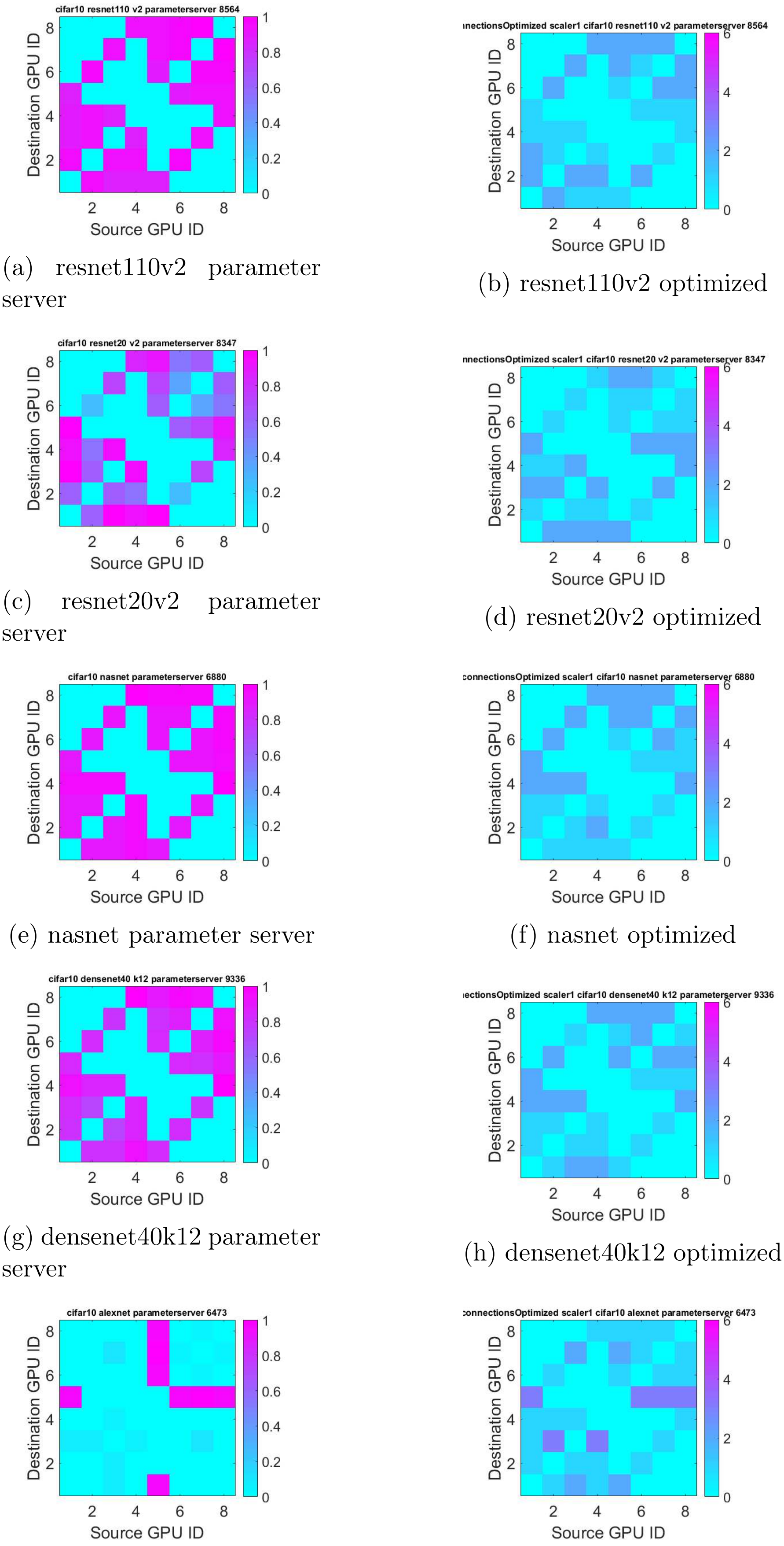
- **Case C:** Figure 4.14 shows the results obtained with Flowers input and parameter server without replication. *Vgg16* has high traffic flow with node 4, and the optimized connectivity show balanced traffic using nodes 3, 5, 6, 7, and 8. *Mobilenet* model has high traffic in node 2, while the optimized connectivity show almost no variation with the relation to the connectivity matrix.

- **Case D:** Figure 4.15 shows the results obtained with Flowers input and parameter server with replication. *Vgg16* shows high traffic in node 3, while the optimized connectivity shows that it can be balanced using nodes 5 and 8.

We made three observations from the bar plot of our listed results, depicted in Figures 4.16a and 4.16b. First, the achieved average reductions in total RTL when optimizing connectivity were: i) 2.52% in Case A, ii) 12.65% in Case B, iii) 9.75% in Case C, and iv) 15.75% in Case D. As expected, the replicated variable models experience increased performance relative to the parameter server models. This is due to higher levels of peer-to-peer traffic as a result of variable synchronization. Second, only a small reduction is seen when the original hyper-mesh topology is well suited to the application. The maximum percentage decrease in RTL using replicated variables was 24.77% using the Flowers training set on the AlexNet model, and the minimum was 1.09% for ResNet110 with Cifar10 input. For parameter server, the maximum was 24.91% for AlexNet with Cifar10 input, and the minimum was 0.52% for NasNet with Cifar10 input. Third, on average, only four additional receivers per GPU are required to optimized traffic. However, this can be dependant on the application communication behavior.

4.6 Final Considerations

We proposed a novel optically-connected GPU architecture that uses an exhaustive minimization algorithm for bandwidth-steering multi-GPU systems. Our reconfigurable architecture efficiently distributes bandwidth between GPU devices showing up to 20% data rate improvement with deep learning applications. Our evaluation shows a significant decrease in measured relative transmission latency (RTL) at the cost of a few additional receivers (Rx) per GPU. Future work will analyze the trade-offs between reduced latency and additional Rx hardware and determine how other applications could benefit from this architecture.



(a) resnet110v2 parameter server

(b) resnet110v2 optimized

(c) resnet20v2 parameter server

(d) resnet20v2 optimized

(e) nasnet parameter server

(f) nasnet optimized

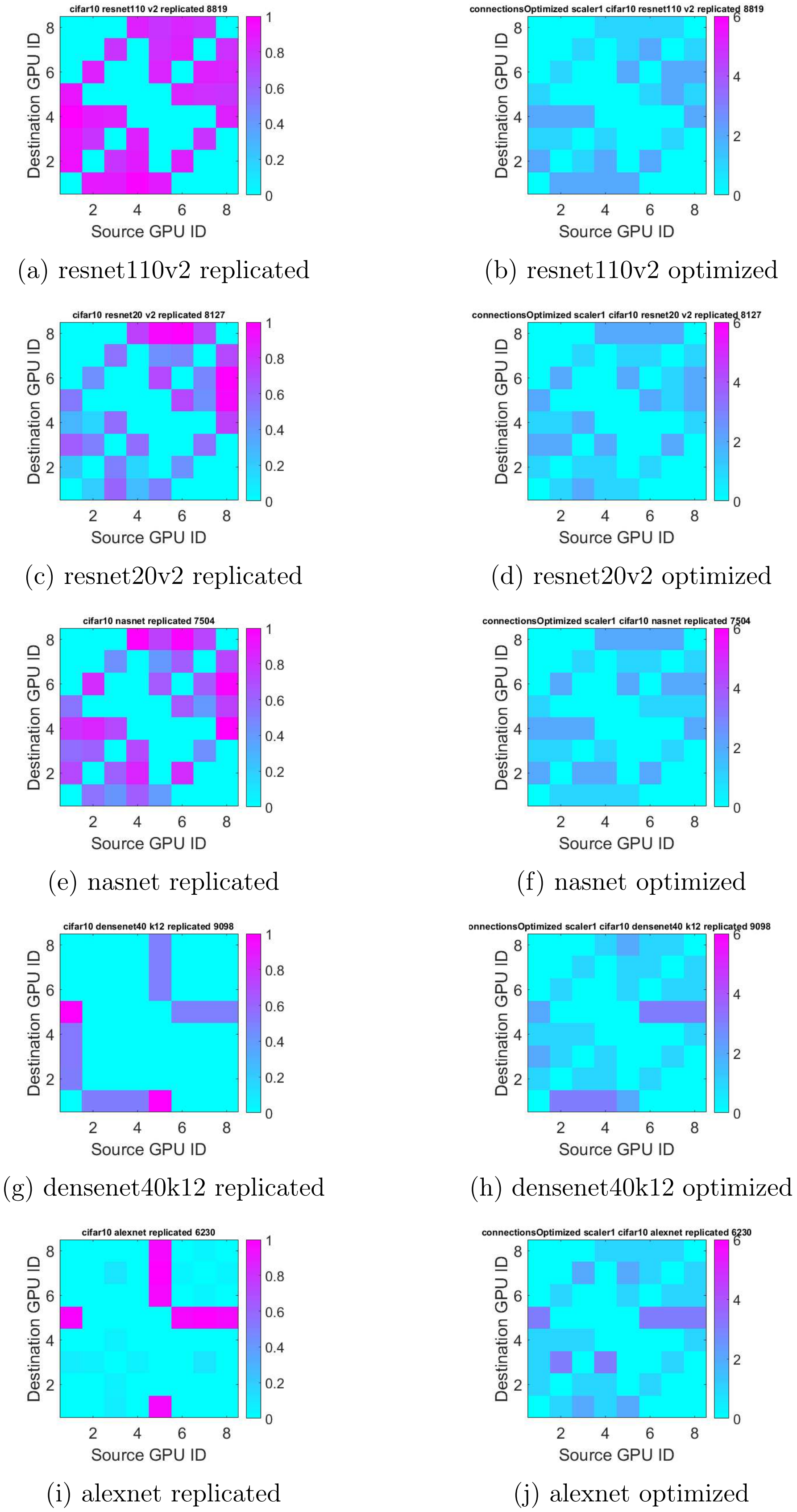
(g) densenet40k12 parameter server

(h) densenet40k12 optimized

(i) alexnet parameter server

(j) alexnet optimized

Figure 4.12: Parameter server with cifar10 input



(a) resnet110v2 replicated

(b) resnet110v2 optimized

(c) resnet20v2 replicated

(d) resnet20v2 optimized

(e) nasnet replicated

(f) nasnet optimized

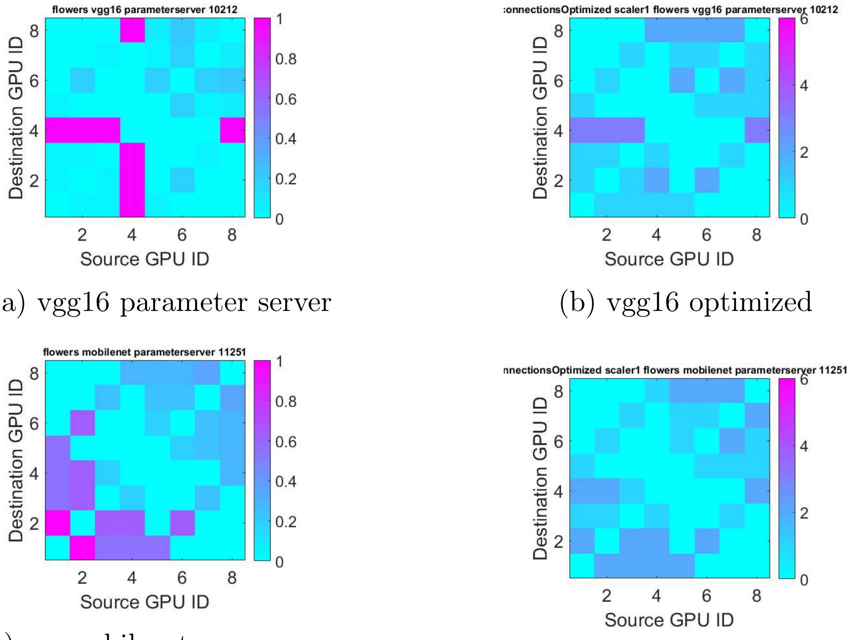
(g) densenet40k12 replicated

(h) densenet40k12 optimized

(i) alexnet replicated

(j) alexnet optimized

Figure 4.13: Replicated with cifar10 input



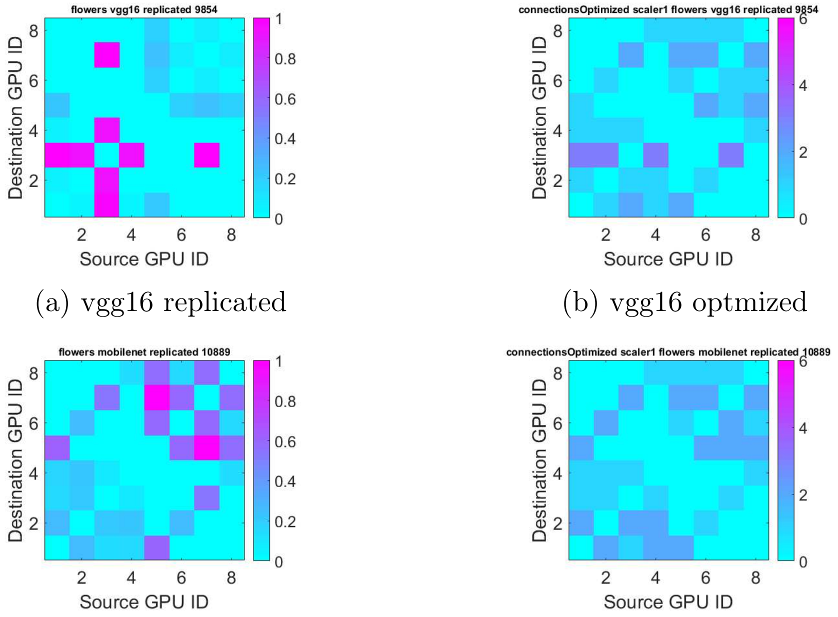
(a) vgg16 parameter server

(b) vgg16 optimized

(c) mobilenet parameterserver

(d) mobilenet optimized

Figure 4.14: Parameter server with flowers input



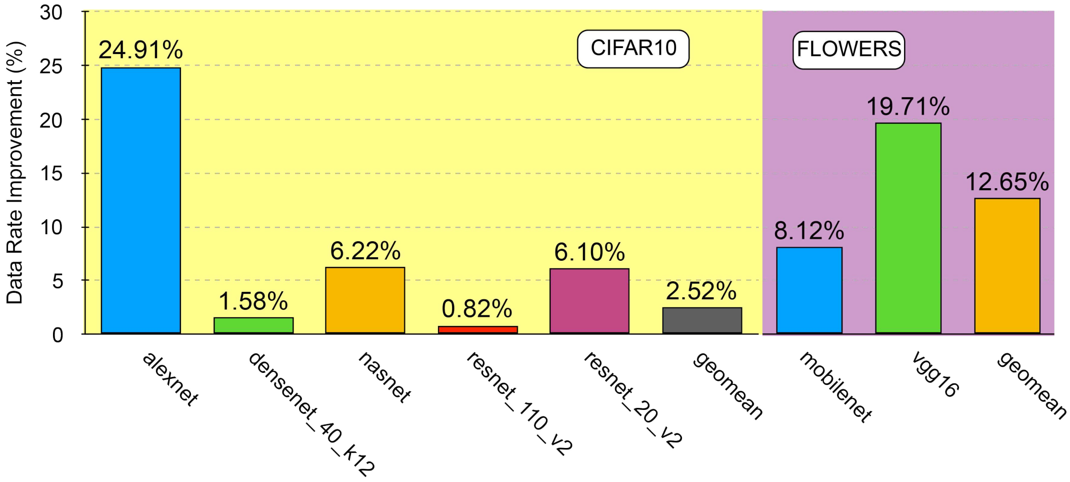
(a) vgg16 replicated

(b) vgg16 optmized

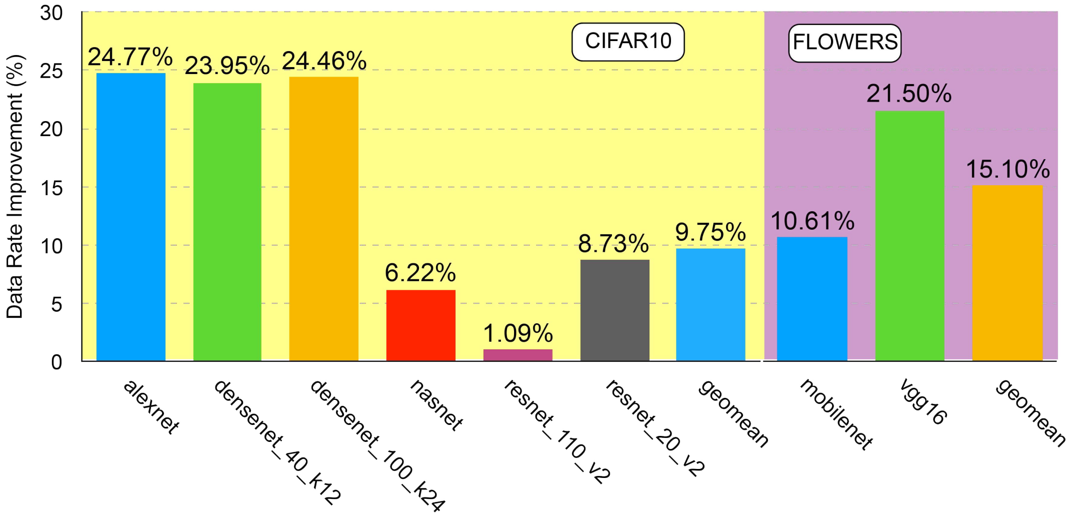
(c) mobilenet replicated

(d) mobilenet optimized

Figure 4.15: Replicated with flowers input



(a) Speedup PS



(b) Speedup PSR

Figure 4.16: Data rate improvement with optical bwsteering

Chapter 5

Optically Connected Memory for Disaggregated Computing

In this chapter, we propose and evaluate an Optically Connected Memory (**OCM**) architecture that disaggregates the main memory from the computation nodes in data centers. OCM is based on micro-ring resonators (MRRs), and it does not require any modification to the DRAM memory modules. In 2020 [95], we calculated energy consumption from real photonic devices and integrate them into a system simulator to evaluate performance. Our results show that (1) OCM is capable of interconnecting four DDR4 memory channels to a computing node using two fibers with 1.07 pJ energy-per-bit consumption and (2) OCM performs up to 5.5× faster than a disaggregated memory with 40G PCIe NIC connectors to computing nodes.

5.1 Main Memory Disaggregation

Scaling and maintaining conventional memory systems in modern data centers is challenging for three fundamental reasons. First, the dynamic memory capacity demand is difficult to predict in the short, medium, and long term. As a result, memory capacity is usually over-provisioned [157, 192, 55, 210, 71], which wastes resources and energy. Second, workloads are limited to using the memory available in the local server (even though other servers might have unused memory), which could cause memory-intensive workloads to slow down. Third, memory maintenance might cause availability issues [171]; in case a memory module fails, all running applications on the node may have to be interrupted to replace the faulty module. A promising solution to overcome these issues is to disaggregate the main memory from the computing cores [146]. As depicted in Figure 5.1, the key idea is to organize and cluster the memory resources such that they are individually addressable and accessible from any processor in the data center [35]. Memory disaggregation provides flexibility in memory allocation, improved utilization of the memory resources, lower maintenance costs, and lower energy consumption in the data center [193].

Disaggregating memory and processors remains a challenge, although the disaggregation of some resources (e.g., storage) is common in production data centers [138]. Electrical interconnections in rack-distances do not fulfill the low latency and high bandwidth

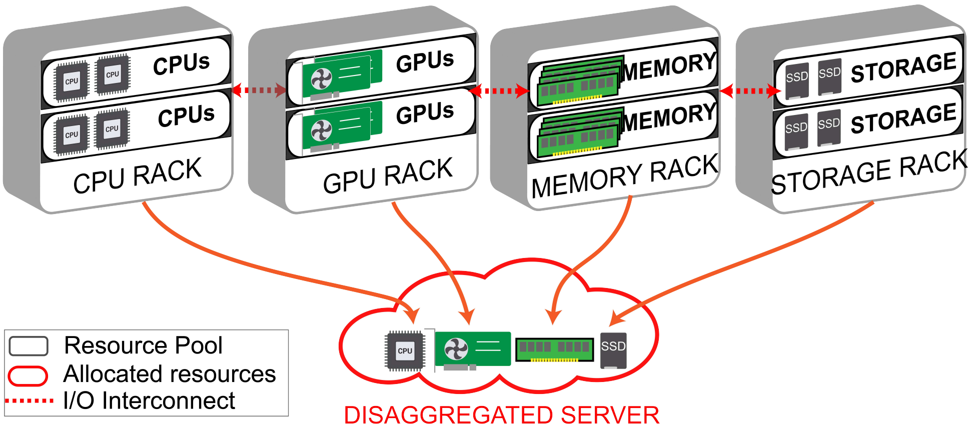


Figure 5.1: Disaggregation concept for data centers.

requirements of modern DRAM modules. The primary limitation of an electrical interconnect is that it constrains the memory bus to onboard distance [247] because the electrical wire's signal integrity loss increases at higher frequencies. This loss dramatically reduces the Signal-to-Noise Ratio (SNR) when distances are large. An optical interconnect is more appealing than an electrical interconnect for memory disaggregation due to three properties: its (1) high bandwidth density significantly reduces the number of IO lanes, (2) power consumption and crosstalk do *not* increase with distance, and (3) propagation loss is low. Silicon Photonic (SiP) devices are likely suitable for disaggregation, delivering $\geq$ Gbps range bandwidth, as well as efficient and versatile switching [61].

In this thesis, our goal is to pave the way for designing high-performance *optical memory channels* (i.e., the optical equivalent of an electrical memory channel) that enable main memory disaggregation in data centers.

Our study provides an optical link design for DDR DRAM memory disaggregation, and it defines its physical characteristics, i.e., i) number of Micro-Ring Resonator (MRR) devices, ii) bandwidth per wavelength, iii) energy-per-bit, and iv) area. We evaluate the performance (see Section 5.4.2) and energy consumption (see Section 5.4.3) of a system with disaggregated commodity DDR DRAM modules.

We make three key contributions: (1) we propose the Optically Connected Memory (OCM) architecture for memory disaggregation in data centers based on state-of-the-art photonic devices, (2) we perform the first evaluation of the energy-per-bit consumption of a SiP link using the bandwidth requirements of current DDR DRAM standards, and (3) we model and evaluate OCM in a system-level simulator and show that it performs up to 5.5x faster than a 40G NIC-based disaggregated memory.

5.2 Bandwidth Scaling for DDR memory

Photonics is very appealing for memory disaggregation because: (1) the integration (monolithic and hybrid) between electronics and optics has already been demonstrated [6], which allows the design and fabrication of highly-integrated and complex optical subsystems on a chip, and (2) optical links offer better scaling in terms of bandwidth, energy, and IO compared to electrical links; e.g., optical switches (o-SW) show better port count scaling [223]).

New electrical interfaces, such as GenZ, CCIX, and OpenCAPI, can disaggregate a wide range of resources (e.g., memory, accelerators) [34]. Optical devices can enable scalable rack-distance, and energy-efficient interconnects for these new interfaces, as demonstrated by a previous work that disaggregates the PCIe interface with silicon photonics [275]. Our OCM proposal extends the memory interface with optical devices and does not require substantial modifications to it, e.g., the memory controllers remain on the compute nodes.

Figure 5.2 shows the IO requirements in the memory controller for electrical [163], and optical interconnects to achieve a specific aggregated bandwidth. We define IO as the number of required electrical wires or optical fibers in the interconnects. We use, for both electrical and optical interconnects, 260-pin DDR4-3200 DRAM modules with 204.8 Gbps maximum bandwidth per memory channel. We make two observations. First, the required number of optical IOs (left y-axis) is up to three orders of magnitude smaller than the electrical IOs because an optical fiber can contain many *virtual channels* using Wavelength Division Multiplexing (WDM) [40, 26]. Second, a single optical IO achieves up to 800 Gbps based on our evaluation, requiring 2 IOs for bidirectional communication (see Section 5.4.3). An optical architecture could reach the required throughput for a 4 memory channel system using only 2 IOs (two fibers) and for a 32-channel system with only 10 IOs.

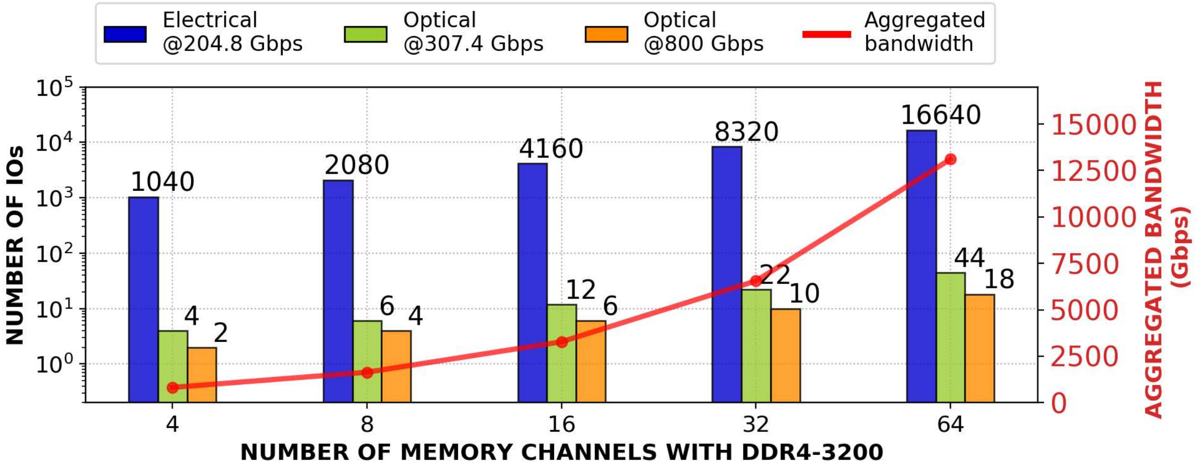


Figure 5.2: Required electrical and optical IO counts (lower is better) for sustaining different amounts of aggregated bandwidth.

5.3 OCM: Optically Connected Memory

To overcome the electrical limitations that can potentially impede memory disaggregation, we introduce an OCM that does not require modifications in the commonly-used DDR DRAM protocol. OCM places commodity DRAM Dual Inline Memory Modules (DIMMs) at rack-distance from the processor, and it sustains multiple memory channels by using different wavelengths for data transmission. OCM uses conventional DIMMs and memory

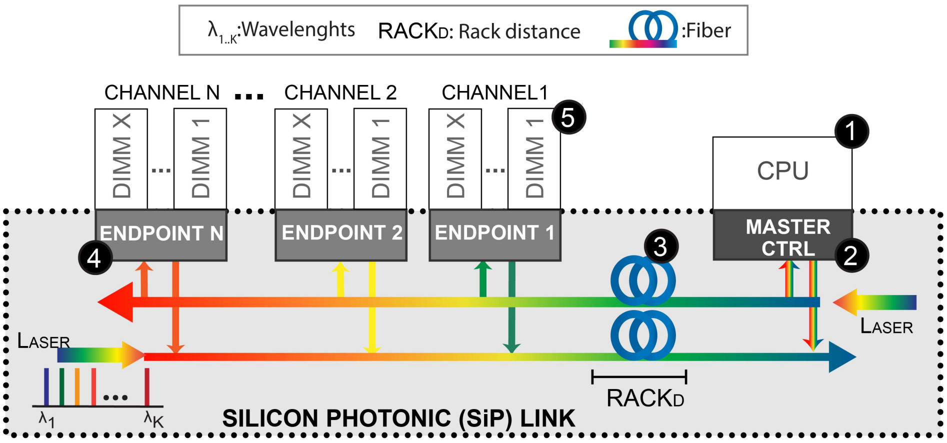
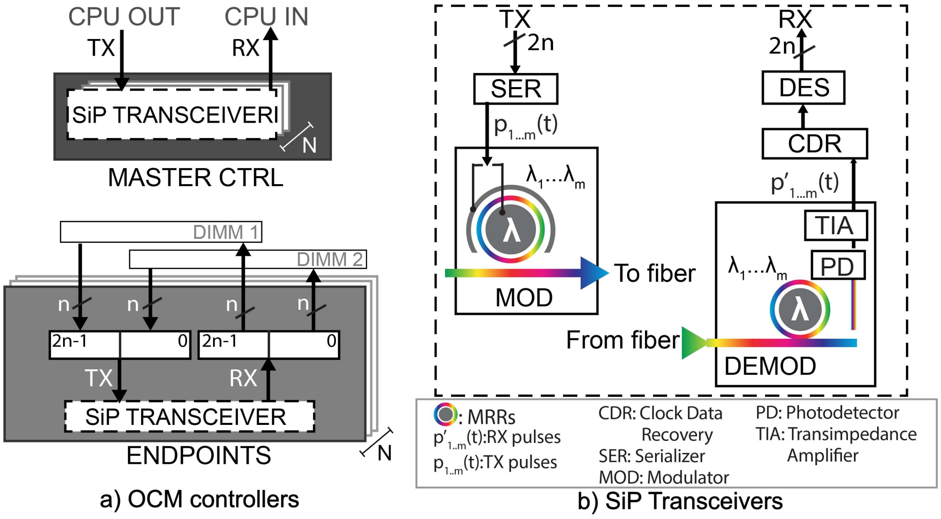


Figure 5.3: Optically Connected Memory organization: optical memory channels for disaggregation of the main memory system.



a) OCM controllers

b) SiP Transceivers

Figure 5.4: Optically Connected Memory organization: optical memory channels for disaggregation of the main memory system.

controllers, electro-optical devices, and optical fibers to connect the computing cores to the memory modules. We explore the idea of direct point-to-point optical interconnects for memory disaggregation and extends prior works [41, 16], to reduce the latency overhead caused by additional protocols such as remote direct memory access (RDMA) and PCIe [271]. Our OCM architecture can scale with the increasing number of wavelengths per memory channel expected from future photonic systems [94].

5.3.1 Architecture Overview

Figure 5.3 and 5.4 show the main components of the OCM architecture configured with state-of-the-art: a) photonic devices such as MRR modulators [24], lasers [156], and photodetectors [26]); and b) DDR4 memories. OCM uses N optical memory channels, each one consisting of X memory modules (DIMM 1 to X) operating in lockstep. OCM uses two key mechanisms to take advantage of the high aggregated bandwidth of the optical domain while minimizing the electrical-optical-electrical conversion latency overhead. First, it implements an optical memory channel with multiple wavelengths that can support multiple DIMMs in a memory channel. Second, it achieves high throughput by increasing the cache line size and splitting it across all the DIMMs in a memory channel. For example, if OCM splits a single cache line between two DIMMs, it halves the bus latency (i.e., data burst duration tBL), compared to a conventional DDR memory.

In our evaluation (Section 5.4), we use two DDR channels operating in lockstep to get a cache line of 128 bytes with similar latency as a cache line of 64 bytes in a single DDR channel (Section 5.3.2). OCM benefits from the use of a wide Xn-bit interface, where X is the number of DIMMs, and n is the width in bits of a DIMM bus. OCM transfers depend on the serialization capabilities of the SiP transceiver. A SERDES allows establishing high data rate communication channels by an encoding/decoding process, taking a parallel input, and transforming it to a serial transmission and vice versa depending on clock recovery. The serialization/deserialization latency increases with the number of DIMMs in lockstep. Notice that, a commercial SERDES link (e.g., [102]) supports serialization up to 256B (i.e., four 64B cache lines). As shown in Figure 5.3, on the CPU side, there is a Master controller, and on the memory side, there are N Endpoint controllers that respond to CPU requests. Both controllers have a structure called SiP Transceiver, and Figure 5.4a) shows a difference in the organization of the SiP transceivers per controller. Figure 5.4b) shows the SiP transceivers present in the Transmitter (TX) and Receiver (RX) lanes in both Master and Endpoint controllers. A TX lane consists of a serializer (SER) and Modulator (MOD) for transmitting data. An RX lane contains a Demodulator (DEMOD), a Clock and Data Recovery (CDR) block, and a Deserializer (DES) for receiving data. Both TX and RX lanes connect with a Xn-bit (e.g., X=2 and n=64 in our evaluation) bus to the Endpoint controller, which forms the bridge between the lanes and the DRAM module.

5.3.2 Timing Model

OCM transfers a cache line as a serialized packet composed of smaller units called *flits*, whose number depends on the serialization capabilities of the SiP transceiver. Figure 5.5 presents the timing diagram of the OCM Read (RD) and Write (WR) operations. For reference, a conventional DDR DRAM memory channel uses 64B cache lines; a data bus transfers each line as 8B data blocks in 8 consecutive cycles, and the 1B Command (CMD) and 3B Address (ADDR) use separate dedicated buses. In OCM, as depicted in Figure 5.5, the cache line is transferred in AB-GH flits. We show OCM timing with a $flit$ size that doubles the width of the memory channel data bus, and is the reason for dividing the cache line between DIMMs 1 and 2 to perform parallel access and decrease latency. OCM splits a single cache line between two DIMMs, which halves the bus latency (i.e., tBL [2]), compared to conventional DDR DRAM memory.

For the RD operation, data A and B are read from different DIMMs to compose a flit (AB). Flit AB serialization and transmission occur after the Master controller receives the CMD/ADDR flit. For the WR operation, the Master controller sends the flit containing data blocks AB immediately after the CMD/ADDR flit. After Endpoint deserialization,

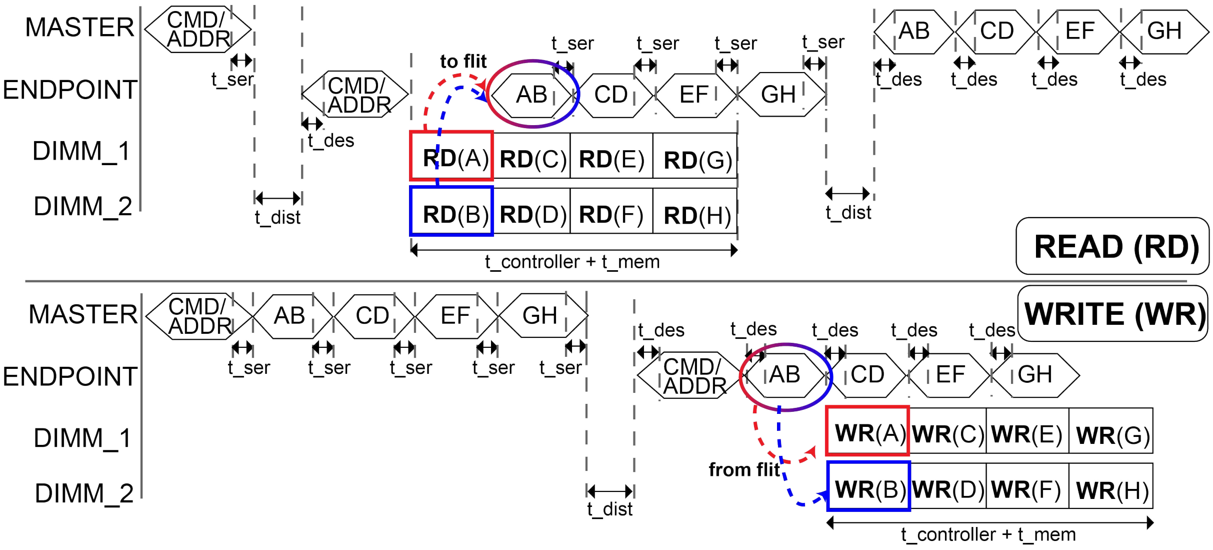


Figure 5.5: OCM timing diagram for Read (top) and Write (bottom) requests.

DIMM 1 stores A, and DIMM 2 stores B. For example, OCM with a commercial Hybrid Memory Cube (HMC) serializer [102] and 128B cache line size, transfers 2 × (4 × 16B of data) with 1 × 4B CMD/ADDR initiator message (or *extra flit*).

Compared to conventional electrical DDR memory, OCM adds serialization and optical packet transport latency to the overall memory access time (see Section 5.4). The DIMM interface can support the latency overhead that is imposed by our optical layer integration. In our evaluation, we consider both optimistic and worst-case scenarios. Past experimental works [16] show that the overhead is low in the order of a few nanoseconds, requiring no modification to the memory controller. However, if there is high latency imposed by the optical layer, the signaling interface from the memory controller needs to be adapted. Equation 5.1 shows the OCM latency model T_{lat}, which is defined as the sum of the DIMM controller latency T_{contr}, DIMM WR/RD latency $T_{mem(A|B)}$ (latency is equal for both DIMMs), serialization/deserialization latency T_{serdes}, modulation/demodulation latencies T_{mod} and T_{demod}, distance propagation latency penalty T_{dist}, and system initialization time (e.g., Clock Data Recovery (CDR) latency, modulator resonance locking [191]) T_{setup}.

$$\begin{aligned} T_{lat}(t) =& T_{setup} + T_{contr} + T_{mem(A|B)}(t) + T_{ser} + T_{des} \\ &+ T_{mod} + T_{demod} + T_{dist} \end{aligned} \tag{5.1}$$

T_{setup} does not affect each transmission because the channel is configured only at boot (or setup), having no impact on the system once it is configured [16]. In the optical and millimeter wavelength bands, T_{mod} and T_{demod} are in the order of ps[26], due to the small footprint of ring modulators (tens of micrometers) and the high dielectric constant of silicon.

5.3.3 Operation

Figure 5.3 illustrates the five stages of a memory transaction.

Stage ❶: processor generates a Read/Write (RD/WR) memory request. In the photonic domain, a laser source generates light in $\lambda_{1,2,\ldots,K}$ wavelengths simultaneously [27].

Stage ❷: data from the processor is serialized (SER) onto the Master Controller's TX lane, and the generated electrical pulses $p_{1,2,\ldots,m}(t)$ drive the cascaded array of Micro-Ring Resonators (MRRs) for modulation (MOD), represented as rainbow rings. We use non-return-to-zero on-off keying (NRZ-OOK) that represents logical ones and zeros imprinted on the envelope of light [26].

Stage ❸: optical signal is transmitted through an optical fiber. At the end of the fiber, the combined optical WDM channels are coupled into an optical receiver.

Stage ❹: first, in the RX lane of an Endpoint, the WDM Demodulator (DEMOD) demultiplexes the optical wavelengths using m MRRs. Each MRR works as an optical band-pass filter to select a single optical channel from $\lambda_{1,2,\ldots m}$. Second, these separated channels are then fed to DEMOD's integrated photo-detectors (PD) followed by transimpedance amplifiers (TIA). Together the PD and TIA convert and amplify the optical signal to electrical pulses $p'_{1,2,\ldots,m}(t)$ suitable for sampling. Third, the data is sampled, deserialized (DES), and sent to the endpoint controller for decoding.

Stage ❺: the processor accesses memory with the DDR protocol using a RD or WR command and a memory address. For a RD command, the Endpoint TX transmits to the processor a *cacheline* with the wavelengths $\lambda_{1,\ldots,m}$ (similar to Stages 1 to 4). For a WR command, the data received from the processor is stored in memory.

5.3.4 Enabling Reconfigurability

OCM supports reconfigurability by placing an o-SW between the Endpoints and the Master controller, similar to previous work [16]. OCM uses optical switching to connect or disconnect a master controller from an endpoint. Switching can happen (1) in the setup phase, which is the first time that the system is connected before starting execution, or (2) before executing a workload, to adapt the amount of assigned memory to the requirements of the workload.

As depicted in Figure 5.6, an optical switch has multiple ports, through which a set of N processors can be connected to a configurable set of M OCMs, where N and M depend on the aggregated bandwidth of the SiP links. In Section 5.4, we evaluate OCM with a single CPU, and assume that the setup phase is already completed.

5.3.5 High Aggregated Bandwidth

OCM uses WDM [26, 40] to optimize bandwidth utilization. WDM splits data transmission into multiple colors of light (i.e., wavelengths, λs).

To modulate data into lightwaves, we use Micro-Ring Resonator (MRR) electro-optical

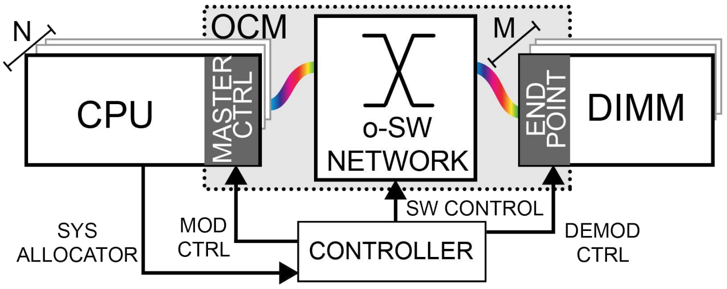


Figure 5.6: Reconfigurable OCM with optical switches (o-SW).

modulators, which behave as narrowband resonators that select and modulate a single wavelength. We use MRRs because they have a small hardware footprint and low power consumption [27], and they are tailored to work in the communications C-band (1530-1565 nm). For more detail on photonic devices, please see [94, 229, 24].

OCM achieves high aggregated bandwidth by using multiple optical wavelengths $\lambda_{1,2,...,K}$ (see laser in Figure 5.3 via WDM in a single link. The K wavelengths are evenly distributed among the controllers, where the TX/RX lanes of a single DDR memory channel have the same number (m) of optical wavelengths ($\lambda_{1,2,...,m}$), see Figure 5.3. All wavelengths have the same bit rate b_r, and the aggregated bandwidth for N memory channels is $BW_{aggr} = b_r \times m \times N$. Assuming that BW_{aggr} is higher than the required bandwidth for a single memory channel BW_{mc}, then $BW_{aggr} = BW_{mc} \times N$. The total number of MRRs is $2 \times 2 \times 2 \times N \times m$ because each TX or RX lane requires m MRRs. OCM has two unidirectional links; each link needs both TX and RX lanes, and these lanes are located in both Endpoint controllers and Master controllers.

5.4 Evaluation

Before showing our evaluations of OCM system-level performance (in Section 5.4.2), and SiP link energy estimation (in Section 5.4.3), we describe our methodology for evaluation.

5.4.1 Evaluation Methodology

OCM performance. To evaluate system-level performance, we implement OCM architecture in the ZSIM simulator [222]. Table 5.1 shows the configuration of our baseline system (a server processor), the two DDR4 memory configurations used in our evaluation (MemConf1 and MemConf2), the latencies of an OCM disaggregated system, and the latencies of a disaggregated system using 40G PCIe NICs. MemConf1 has 4 DDR4 memory channels as in conventional server processors, and MemConf2 has a single DDR4 memory channel, and an in-package DRAM cache on the processor side.

The goal of the DRAM cache is to reduce the optical disaggregation overhead [271], which can have a significant performance impact in memory-bound applications. Our DRAM cache resembles the Banshee DRAM cache [269] that tracks the contents of the DRAM cache using TLBs and page table entries, and replaces pages with a frequency-based predictor mechanism. We configure our DRAM cache to have the same operation latency as commodity DDR4 memory.

Table 5.1: Baseline processor, memory, OCM, and NIC.

Baseline	*Processor*	3 GHz, 8 cores, 128B cache lines
	Cache	32KB L1(D+I), 256KB L2, 8MB L3
MemConf1	*Mem*	4 channels, 2 DIMMs/channel, DDR4-2400 [2]
MemConf2	*Mem*	1 channel, 2 DIMMs/channel, DDR4-2400
	DRAM cache	4GB stacked, 4-way, 4K pages, FBR [269], DDR4-2400
OCM	*SERDES*	latency: 10/150/340 cycles
	Fiber	latency: 30/60/90 cycles (2/4/6 meters roundtrip)
NIC	*40G PCIe [182]*	latency: 1050 cycles

We calculate the SERDES link latency values for the upcoming years. We estimate the minimum at 10 cycles, which assumes 3.2 ns serialization/deserialization latency [126]. We use 340 cycles (113ns) maximum latency reported in a previously demonstrated optical interconnection system [204]. We simulate rack distances of 2m, 4m, and 6m with a 5 ns/m latency [4], which translates into 30, 60, and 90 cycles latency in our system.

For the 40G NIC-based system configuration, we evaluate a scenario using a PCIe Network Interface Card (NIC) latency of 1050 cycles (350 ns) [4] (a realistic NIC-through-PCIe latency is in the order of microseconds because of the PCIe protocol overhead latency [182]).

We evaluate the system-level performance of OCM with applications from six benchmark suites representing three workload scenarios: (1) multi-program, (2) multithread, and (3) multinode.

1. The first scenario for **multi-programmed workloads** depicts a mix of benchmark applications executing concurrently. We used SPEC06 [107] with Pinpoints (warmup of 100 million instructions, and detailed region of 30 million instructions), and SPEC17 [44] *speed* with reference inputs. Table 5.2 lists the content of the used SPEC benchmark mixes.

Table 5.2: Evaluated SPEC06 & SPEC17 benchmark mixes.

SPEC06	mix1	soplex_1, h264, gobmk_3, milc, zeusm, bwaves, gcc_1, omnetpp
	mix2	soplex_1, milc,povray, gobmk_2, gobmk_3, bwaves, calculix, bzip2_2
	mix3	namd, gromacs, gamess_1, mcf, lbm, h264_2, hmmer, xalancbmk
SPEC17	mix1	exchange2, cactus, gcc_2, imagick, fotonik3d, xalancbmk, xz_2, lbm
	mix2	gcc_1, nab, lbm, leela, mcf, xz_1, sroms, omnetpp
	mix3	xalancbmk, nab, cactus, mcf, imagick, xz_1, fotonik3d, deepjeng

2. The second scenario represents **multithreaded workloads,** i.e., a single application with multiple threads executing on a multicore processor. We used PARSEC [37] with *native* inputs, SPLASH2 [36] with *simlarge* inputs, and GAP graph benchmarks [31] executing 100 billion instructions with the *Web* graph input, and

30 billion instructions with the *Urand* graph input. The *Urand* input has very poor locality between graph vertices compared to the *Web* input. We also used five MPI applications from the NAS Parallel Benchmark (NPB) [28] with class C inputs, executing 100 billion instructions for FT, MG and CG, and whole running IS and EP. For these MPI applications, we split them over 8 processes, 1 process per core.

3. For **multinode workloads**, a single application executes on multiple nodes of a computer cluster. We considered a computer cluster composed of eight nodes with the same characteristics as shown in Table 5.1. We used the same MPI applications from NPB that we used on the multithreaded workload scenario, but instead of running all eight processes on the same node, we distribute them among eight nodes, one process per node. To consider the network overhead from node to node communication, we used a two-step simulation approach. On the **first step**, we execute the MPI application on ZSIM, considering only one of the eight created processes. This execution allows measuring performance, i.e., speedup, without node-to-node communication overhead. On the **second step**, we modeled a computer cluster with the SimGrid simulator [49] and tuned each node's processing power according to the speedup obtained in the first step. We executed the MPI benchmarks on our tuned cluster model using the SimGrid MPI interface [69], obtaining a performance measurement that considers the network overhead. The cluster model we used resembles the topology from a local computer cluster named Kahuna, where the eight nodes are connected via a Mellanox SX6025 switch. We measured the bandwidth and latency on node to node communication in Kahuna, using two kernels, *osu_latency* and *osu_bw*, both from the OSU benchmark suite[148]. In the first step of our two-step simulation approach, we executed 9 billion instructions for IS, 24 billion instructions for FT, 8 billion instructions for CG, 7 billion instructions for MG and 21 billion instructions for EP. On the second step, all the applications executed without any limitation on the number of instructions.

Table 5.3 summarizes the measured memory footprint values for all the benchmarks used in our evaluation, measured using the Massif tool from Valgrind [181]. The measured memory footprint of MPI applications from NPB is for the application code only, and it does not include the memory footprint from the MPI process manager.

We also used a synthetic benchmark, that resembles the *copy* kernel from the STREAM benchmark [166], to obtain the OCM memory roofs, based on the memory roof concept of the Roofline model [258].

SiP link energy-per-bit. To evaluate the interconnection between processor and memory as a point-to-point SiP link, we use PhoenixSim [216] with parameters extracted from state-of-the-art optical devices [26, 201, 23]. Figure 5.7 shows the graphical user interface of the PhoenixSim simulator.

PhoenixSim considers the physical features of the optical devices and their digital semiconductor drivers to evaluate many SiP link energy-per-bit cases in terms of: (1) the required number of optical wavelengths (λ), and (2) the bit rate per λ. Table 5.4 lists OCM optical devices and their main characteristics used in our simulation model.

Table 5.3: Measured memory footprints.

SPEC06 [107]	*MIX1*: 2.2 GB, *MIX2*: 3.1 GB, *MIX3*: 2.4 GB
SPEC17 [44]	*MIX1*: 19.9 GB, *MIX2*: 36.4 GB, *MIX3*: 34.7 GB.
PARSEC [37]	*canneal*: 716.7 MB, *streamcluster*: 112.5 MB, *ferret*: 91.9 MB, *raytrace*: 1.3 GB, *fluidanimate*: 672 MB
SPLASH [36]	*radix*: 1.1 GB, *fft*: 768.8 MB, *cholesky*: 44.2 MB, *ocean_ ncp*: 26.9 GB, *ocean_ cp*: 891.8 MB.
GAP [31]	*Urand* graph: 18 GB, *Web* graph: 15.5 GB
NPB [28] Class C	*Integer Sort (IS)*: 2.3 GB, *Fast Fourier Transform (FFT)*: 7.2 GB, *Conjugate Gradient (CG)*: 1.2 GB, *Multi Grid (MG)*: 3.5 GB, *Embarrassingly Parallel (EP)*: 34.4 MB

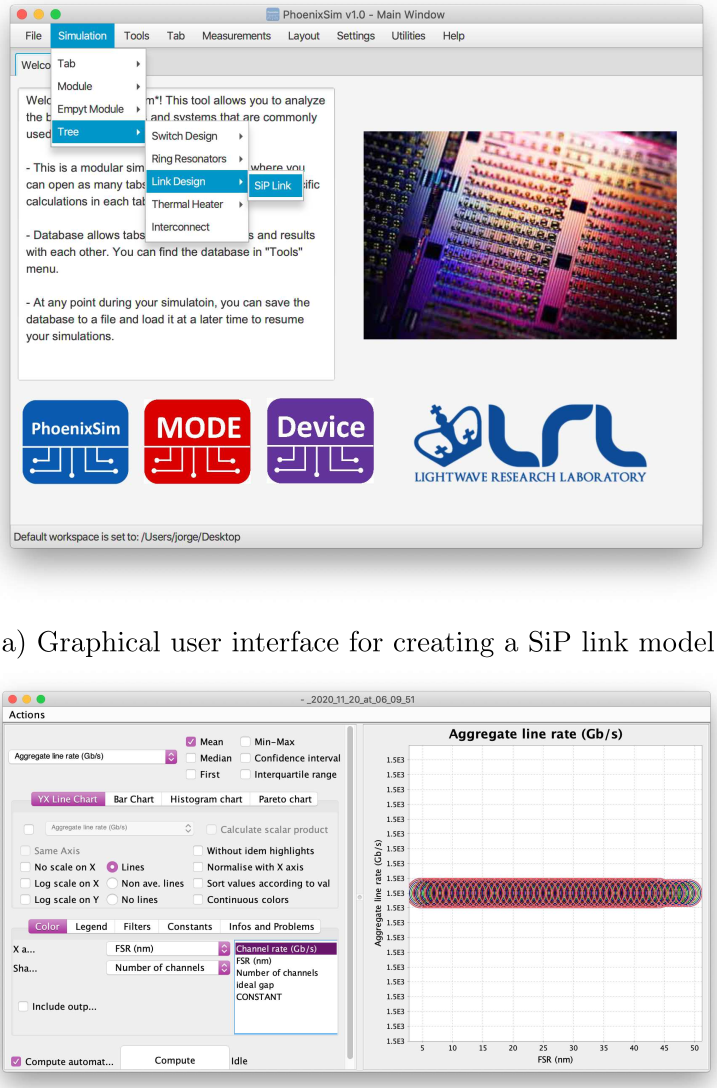
5.4.2 System-level Evaluation

Multiprogrammed Evaluation. Figure 5.8 shows the slowdown of OCM and 40G NIC-based disaggregated memory systems with MemConf1, compared to a non-disaggregated MemConf1 baseline, for three mixes of SPEC06 benchmarks (Table 5.2). Notice that a system with disaggregated main memory is expected to perform worse than the non-disaggregated baseline, because of the extra latency introduced by the interconnects (see Eq. 5.1).

We make two observations. First, the 40G NIC-based system is significantly slower than our OCM system, even though the Ethernet configuration we evaluate is very optimistic (350 ns average latency, equivalent to 1050 cycles in Table 5.1). OCM is up to 5.5× faster than 40G NIC for the minimum SERDES latency, and 2.16× faster for the maximum SERDES latency. Second, the results show the feasibility of low-latency disaggregation with OCM as future SERDES optimizations become available. OCM has an average slowdown (across all rack-distances) of only 1.07× compared to the baseline with a SERDES latency of 10 cycles, and 1.78× average slowdown with a SERDES latency of 340 cycles.

Figure 5.9 shows the speedup of a disaggregated OCM system (green bars) compared to a non-disaggregated baseline, both configured with MemConf1. Figure 5.9 also shows the speedup of OCM with MemConf2 (red bars), and the speedup of a non-disaggregated system with MemConf2 (blue bars), both compared to a MemConf2 baseline without a DRAM cache and without disaggregation. OCM has a conservative SERDES latency of 150 cycles, and a distance of 4m.

Figure 5.9 (left) shows the results for SPEC17 mixes (see Table 5.2). We make two observations. First, the average slowdown of OCM without DRAM cache (green bars) is 17%, which is in the same order as the SPEC06 results (Figure 5.8). Second, with a DRAM cache, the performance of the OCM disaggregated system (red bars), and the non-disaggregated system (blue bars) is very close, as the memory intensity of these benchmarks is not very high. As expected, the performance of the disaggregated system is always lower than the non-disaggregated system.



Welc... m*! This tool allows you to analyze the b... and systems that are commonly used

- This is a modular sim... where you can open as many tabs ...ific calculations in each tab

- Database allows tabs ...s and results with each other. You can find the database in "Tools" menu.

- At any point during your simulatoin, you can save the database to a file and load it at a later time to resume your simulations.

Default workspace is set to: /Users/jorge/Desktop

(a) Graphical user interface for creating a SiP link model.

(b) Simulation output window.

Figure 5.7: PhoenixSim [216] simulator developed by the Lightwave Research Laboratory at Columbia University.

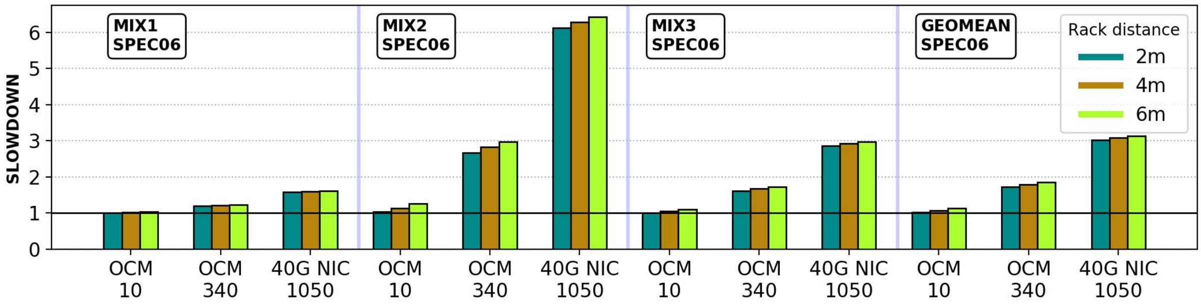


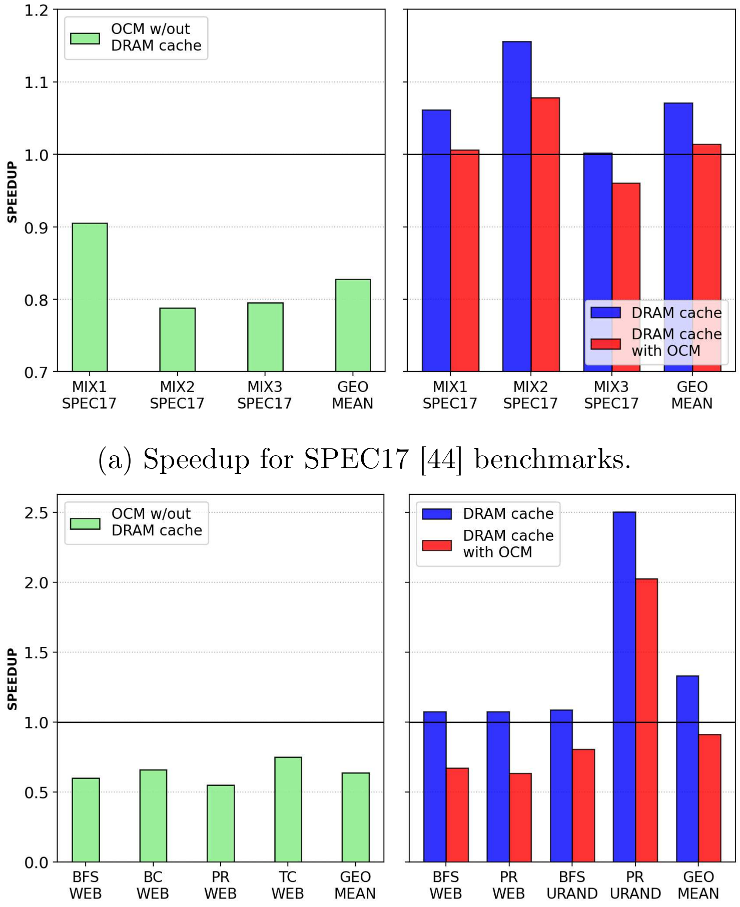
Figure 5.8: Slowdowns of OCM and 40G NIC-based disaggregated systems, compared to a non-disaggregated baseline with MemConf1, for three randomly-selected mixes of SPEC06 benchmarks (lower is better).

Table 5.4: Optical and electrical models for OCM SiP link devices

Parameter	Design Criteria	Details	Ref.
Optical power	20 dBm	Max. aggregated	
Center wavelength	1.55 μm		
Laser	10% and 30%	Laser wall-plug efficiency	[45]
Waveguide loss	5 dB/cm	fabrication roughness	[98]
	0.02 dB/bend	waveguide bend loss	
Coupler loss	1 dB	off-chip coupler	[57]
Modulator	Q = 6500	Ring resonator Q factor	[201]
	ER = 10 dB	MRR extinction rate	
	65 fF	Junction capacitance	
	-5 V	Maximum drive voltage	
	1 mW	Thermal-tuning power/ring	[24]
Mod. mux and demux	MRR power penalties	Crosstalk model	[26]
Photodetector	1.09 A/W	Current per opt. power	[81]
Modulator driver	28 nm	Semicond. tech. for OOK-WDM	[201]
SERDES power model	28 nm	Semicond. tech.	[201]
Digital receiver	28 nm	Semicond. tech. for OOK-NRZ	[201]
Element positioning	100 μm	Modulator padding	

Multithreaded Evaluation. Figure 5.9 (right) shows the results for multithreaded graph applications. We make two observations. First, the maximum slowdown of OCM without a DRAM cache (green bars) is up to 45% (*pagerank* (*PR*)), which is in the same order as SPEC17 results, despite the *Web* input having very high locality. The extra latency of the OCM disaggregated system has a clear negative effect on performance. Second, graph workloads dramatically benefit from using a DRAM cache (red and blue bars), e.g., *PR* with *Urand* input shows a speedup of 2.5× compared to the baseline, which is 50% lower speedup than the non-disaggregated scenario. We believe that the performance degradation of OCM with DRAM cache is still reasonable. However, adding a DRAM cache also brings new challenges that need further investigation in a disaggregated setting, such as page replacement mechanisms and caching granularity [269, 145, 170, 267, 172, 266, 207, 117].

Figure 5.10 shows the slowdown of OCM compared to the baseline, using MemConf1 with PARSEC and SPLASH2 benchmarks. We show results for the memory-bound benchmarks only. We also test other compute-bound benchmarks that show less than 5% slowdown, as depicted in Figure 5.13 and 5.12. We make three observations. First, with the lower bound SERDES latency (10 cycles) and lowest rack distance (2 m), applications such as *streamcluster*, *canneal* and *cholesky*, experience an average 3% speedup. This small improvement occurs as a result of T_{mem} reduction (tBL related) due to splitting of a cache line into two DIMMs. Second, the slowdowns increase slightly as distance increases. Third, with large rack-distance and maximum SERDES latency, the slowdown is significant. The highest slowdown measured is 2.97× for *streamcluster* at 6m and 340 SERDES cycles; the average slowdown is 1.3× for SPLASH2 and 1.4× for PARSEC.



(a) Speedup for SPEC17 [44] benchmarks.

(b) Speedup for GAP [31] graph benchmarks.

Figure 5.9: OCM speedup results with 4m distance and a SERDES latency of 150 cycles (higher is better), compared to a disaggregated baseline, with or without a DRAM cache

Figure 5.11 shows the slowdown of OCM with DRAM cache in a conservative scenario, i.e., medium rack distance (4m) and SERDES latency (150 cycles), using MemConf2 with memory-bound benchmarks of PARSEC and SPLASH2. We additionally explore MemConf2 with a 1 GB DRAM cache. We make two observations. First, using a DRAM cache reduces the latency overhead caused by OCM disaggregation. The average slowdown is 0.89× for OCM with a 1 GB DRAM cache and 0.83× for OCM with a 4 GB DRAM cache. However, OCM with a 1 GB and 4GB DRAM cache performs faster compared to the 1.33× slowdown of an OCM system without DRAM cache. Second, OCM can also benefit from a lowersized DRAM cache depending on the workload memory footprint and access behavior. The *ocean_ncp* and *streamcluster* benchmarks have the highest slowdown with OCM. Both benchmarks exhibit a similar performance improvement using a 1 GB DRAM cache compared to a 4 GB DRAM cache. The *ocean_ncp* benchmark performs only 3% slower in an OCM system with a 1 GB DRAM cache than an OCM system with a 4 GB DRAM cache. While executing *ocean_ncp* benefits from a 1 GB DRAM cache because of its large memory footprint of ≈ 27 GB, benchmarks with low memory footprint such as *cholesky* (≈ 44 MB) does not benefit from a DRAM cache due to the TLB overhead. Using a smaller DRAM cache can help reduce area and electrical

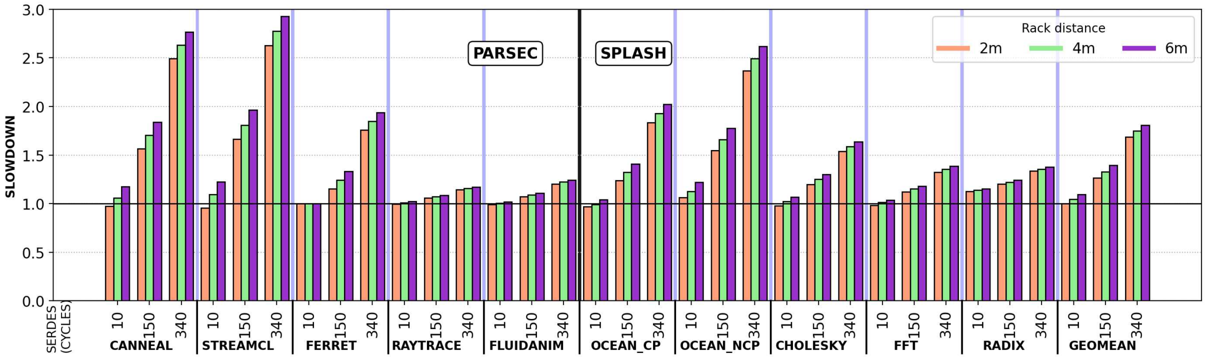


Figure 5.10: OCM slowdown compared to the baseline for PARSEC and SPLASH2 benchmarks (lower is better).

energy consumption on an OCM system's processing side.

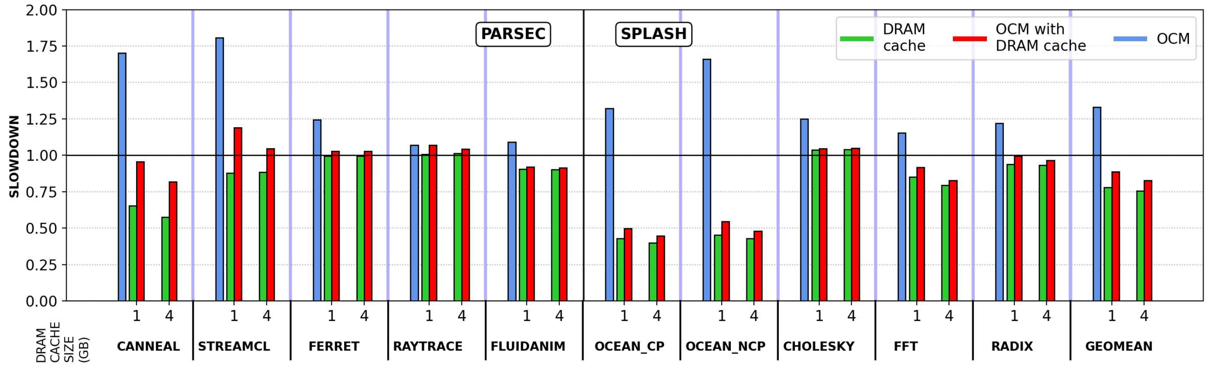


Figure 5.11: OCM slowdown results with a DRAM cache for PARSEC and SPLASH2 benchmarks on a system with 150 SERDES latency and 2m rack distance (lower is better).

Figure 5.14 shows the OCM speedup using MemConf2 with the NPB benchmarks. These results present a maximum slowdown of 27% with the CG benchmark, while with the other benchmarks, the performance loss stays within 14%. Using a DRAM cache exhibits a performance improvement on all benchmarks except on EP. This occurred due to the extremely low memory footprint from this benchmark, as depicted in Table 5.3.

Multinode Evaluation. Figure 5.15 shows the results of a multinode scenario running NPB benchmarks on eight different nodes, one process per node. This multinode execution case exhibits a reduced variation in performance compared to the multithreaded workloads. The difference between the average performances of the three configurations stays within 7%. This is due to two factors that diminish the impact of the memory system. The first factor is that the memory footprint of each benchmark was also split among the eight nodes. Considering the 34.4 MB memory footprint from EP, the eighth part is around 4.3 MB, which entirely fits the 8 MB L3 Cache. The second factor is that network performance has a significant impact on these applications. We considered a node to node latency of 8 microseconds (around 24000 cycles). As a comparison, our worst SERDES overhead consideration for OCM was 113 nanoseconds (340 cycles). Depending on how the applications can overlap computation and network communication, the performance bottleneck may shift from the memory system to the network performance.

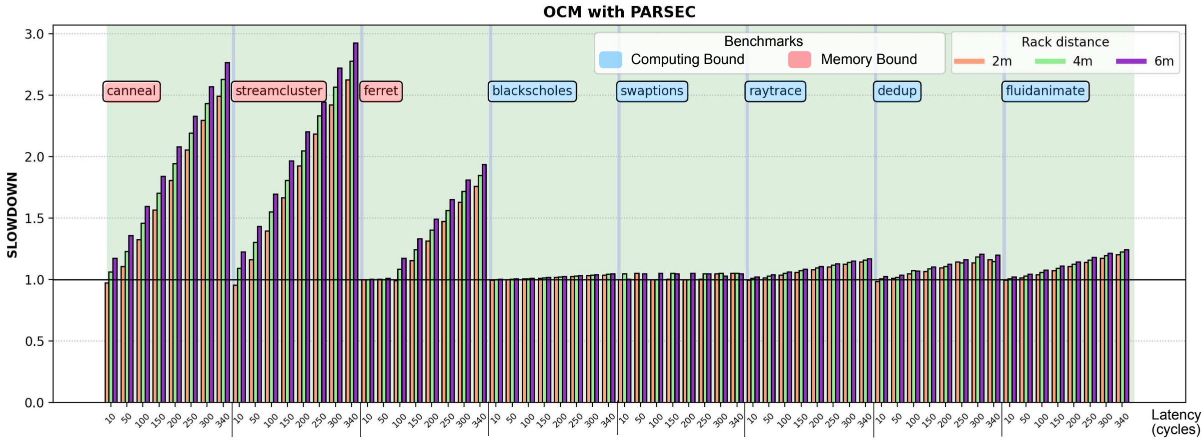


Figure 5.12: OCM slowdown for PARSEC benchmarks. The applications are grouped as computing-bound (blue) and memory-bound (red).

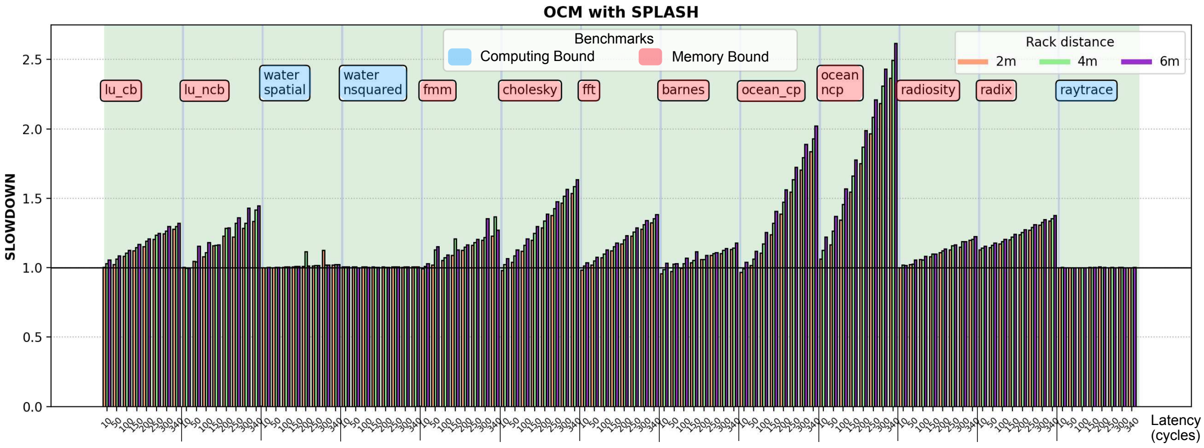


Figure 5.13: OCM slowdown compared for SPLASH2 benchmarks. The applications are grouped as computing-bound (blue) and memory-bound (red).

Memory Roofs. Figure 5.16 and 5.17 present the memory roofs obtained with MemConf2 configuration. Figure 5.16 represents the memory roofs obtained from a single core (1 thread), while Figure 5.17 represents the memory roofs obtained from a multithreaded execution (8 threads, one per core). OCM with DRAM cache shows an increase in bandwidth performance according to the bandwidth demand. They increase their bandwidth on the multithreaded case (3.79× for DRAM cache and 1.88× for OCM), while the electrical memory exhibits no variation on its performance. With the higher bandwidth demand from the multithreaded application, OCM compares to the electrical memory in bandwidth performance, and the DRAM cache exhibits an advantage over the electrical memory. This behavior shows that OCM can achieve similar performance to the electrical memory bandwidth on the best case (cache-friendly memory accesses). Concurrently, a DRAM cache may become only an additional level on the memory hierarchy, without any gain of performance, on lower bandwidth demands.

We conclude that OCM is very promising because of its reasonably low latency overhead (especially with the use of a DRAM cache), and the flexibility of placing memory modules at large distances with small slowdowns.

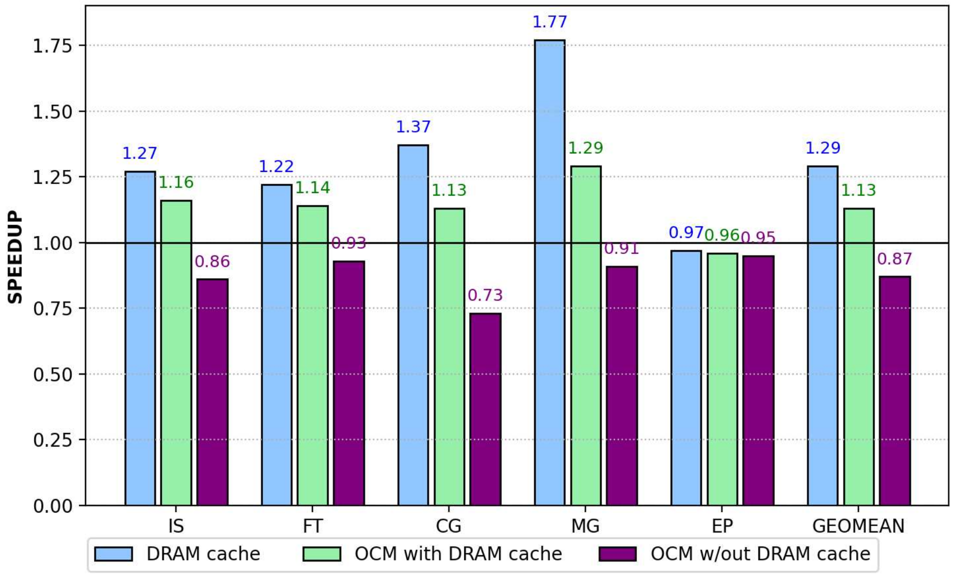


Figure 5.14: Speedup of the usage of DRAM cache (with and without OCM) and OCM compared to the baseline for NPB benchmarks (higher is better), using eight processes all on a single node.

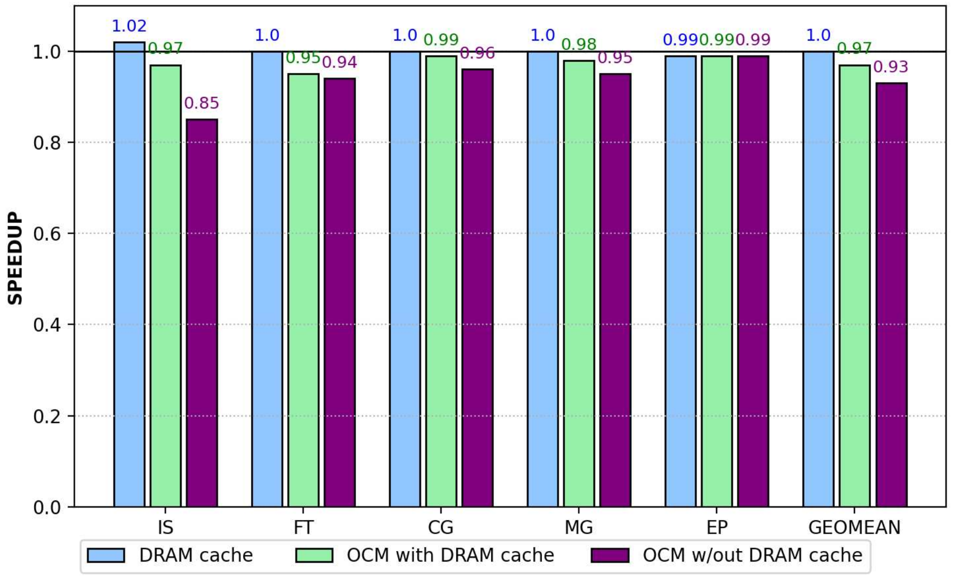


Figure 5.15: Speedup of the usage of DRAM cache (with and without OCM) and OCM compared to the baseline for NPB benchmarks (higher is better), using eight processes distributed among eight nodes (1 process per node).

5.4.3 SiP Link Evaluation

We evaluate the energy and area consumption of the SiP link to allow the system designer to make tradeoffs about the use of SiP devices in the computing system. It is enough to consider a single unidirectional modelled SIP link using PhoenixSim [216] with the input parameters shown in Table 5.4 to estimate the energy efficiency. We estimate the minimum energy-per-bit consumption and the required number of MRRs for our model, given an aggregated optical bandwidth equivalent to the bandwidth required by DDR4-

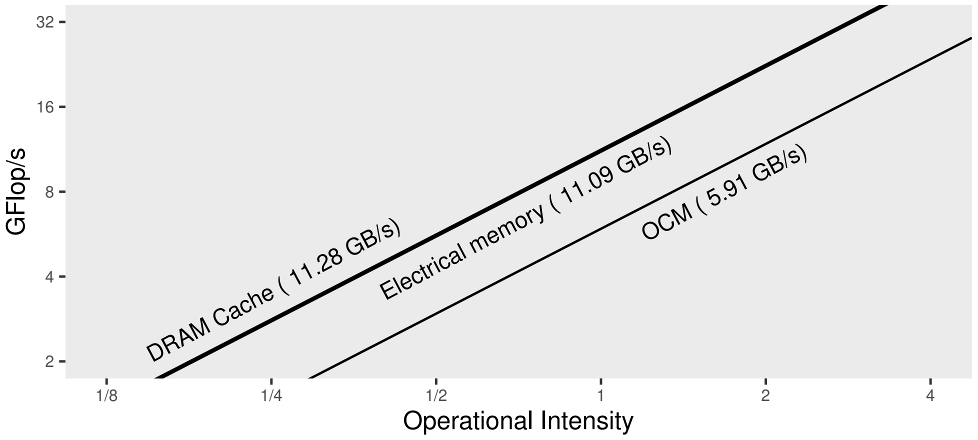


Figure 5.16: Memory roof, using MemConf2, of DRAM cache, electrical memory and OCM using a single threaded application.

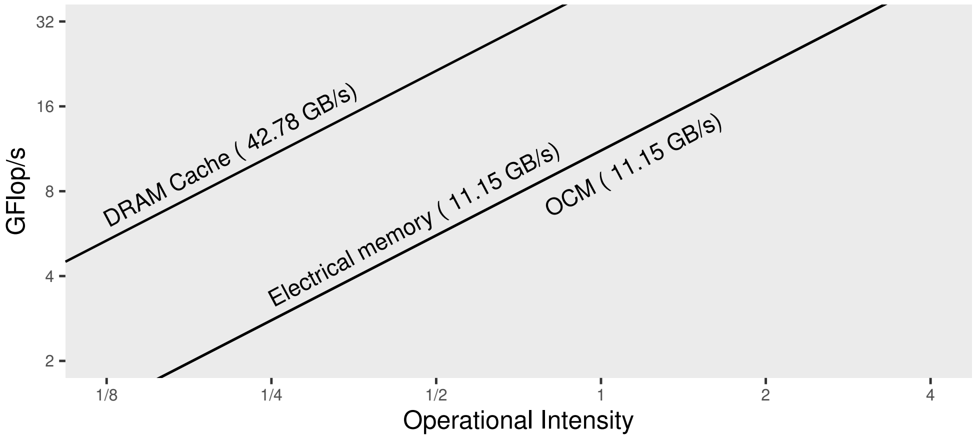


Figure 5.17: Memory roof, using MemConf2, of DRAM cache, electrical memory and OCM using a multithreaded application (8 threads).

2400 DRAM memory.

A single DDR-2400 module requires 153.7 Gbps bandwidth [2]. 4 memory channels, with 2 DIMMs per channel in lockstep, require ∼615 Gbps/link. OCM's maximum feasible bandwidth (while remaining CMOS compatible) is 802 Gbps using the parameters in Table 5.4. More advanced modulation formats, such as PAM4 [229], can be used to achieve higher aggregated bandwidth. Figures 5.18 and 5.19 show the energy-per-bit results (y-axis), and the aggregated bandwidth. The aggregated link bandwidth is the multiplication of the number of λ (bottom x-axis values), and the aggregated bitrate (top x-axis values), i.e., a higher number of λs implies a lower bitrate per λ. We consider three feasible and efficient MRR sizes in our model: 156.4 (green), 183.5 (orange), and 218.4 μm^2 (blue).

From Table 5.4, we have considered two cases of lasers, 10%-efficient epitaxially-grown integrated laser, which is widely used in the SiP industry [124], and a state-of-the-art laboratory laser with a nominal efficiency of 30% [45] to demonstrate that improvement of optical features of a single device affects our SiP link energy estimation significantly. Our previous work [95] used off-chip lasers, while in this work, we report results using heterogeneous integration of lasers on silicon [124] and reducing the number of couplers per link.

As shown in Figure 5.18, in OCM with 615 Gbps links using lasers with 10% efficiency, the minimum energy consumption overhead compared to the electrical memory system is 1.02 pJ/bit for 35 optical wavelengths (λ) per link, each λ operating at 17.57 Gbps. The SiP link with the 30% laser efficiency achieved and energy consumption of 0.64 pJ/bit with 39 λ's, each operating at 15.8 Gbps, as depicted in Figure 5.19.

The energy evaluation of the maximum feasible bit rate of a SiP link is also presented, with an aggregate bandwidth of 800 Gbps. The minimum energy consumption is 1.43 pJ/bit for 36 λ's per link, each λ operating at 22.22 Gbps using a laser with 10% efficiency. The SiP link that has lasers with an efficiency of 30% showed energy consumption of 0.81 pJ/bit for 45 λ's, each operating at 17.77 Gbps.

We make three observations from Figures 5.18 and 5.19. First, as in electrical systems, it is expected that a higher bandwidth per link increases the link energy-per-bit consumption. However, the optical energy-per-bit is lower compared to electrical systems. For reference, the energy-per-bit of a DDR4-2667 DRAM module is 39 pJ [194]; thus, the energy-per-bit caused by an additional SiP link in the memory subsystem is less than 5%. Second, there is a non-smooth behavior on the energy-per-bit curves due to the energy consumption model of the optical receiver, which depends on the data rate. In our model, we set the photodetector current to a minimum value. As the data rate increases, the received signal becomes less distinguishable from noise. Our model forces the photocurrent to step into a new minimum value to avoid this, causing the repeated decrease and increase of the energy-per-bit values [27]. For both SiP links, the 183.5 μm^2 rings consume the lowest energy. The estimated area overhead is 51.4E-3 mm^2 with 2 × 615 Gbps links, and 57.3E-3 mm^2 with 2×802 Gbps links. In our case study of 4 DDR4 memory channels, OCM uses fewer physical interconnects (optical fibers) than 40G PCIe NIC links (copper cables). In other words, to achieve the required aggregated link bandwidth, we require 2 optical fibers with OCM or 30 copper cables with 40G PCIe NICs.

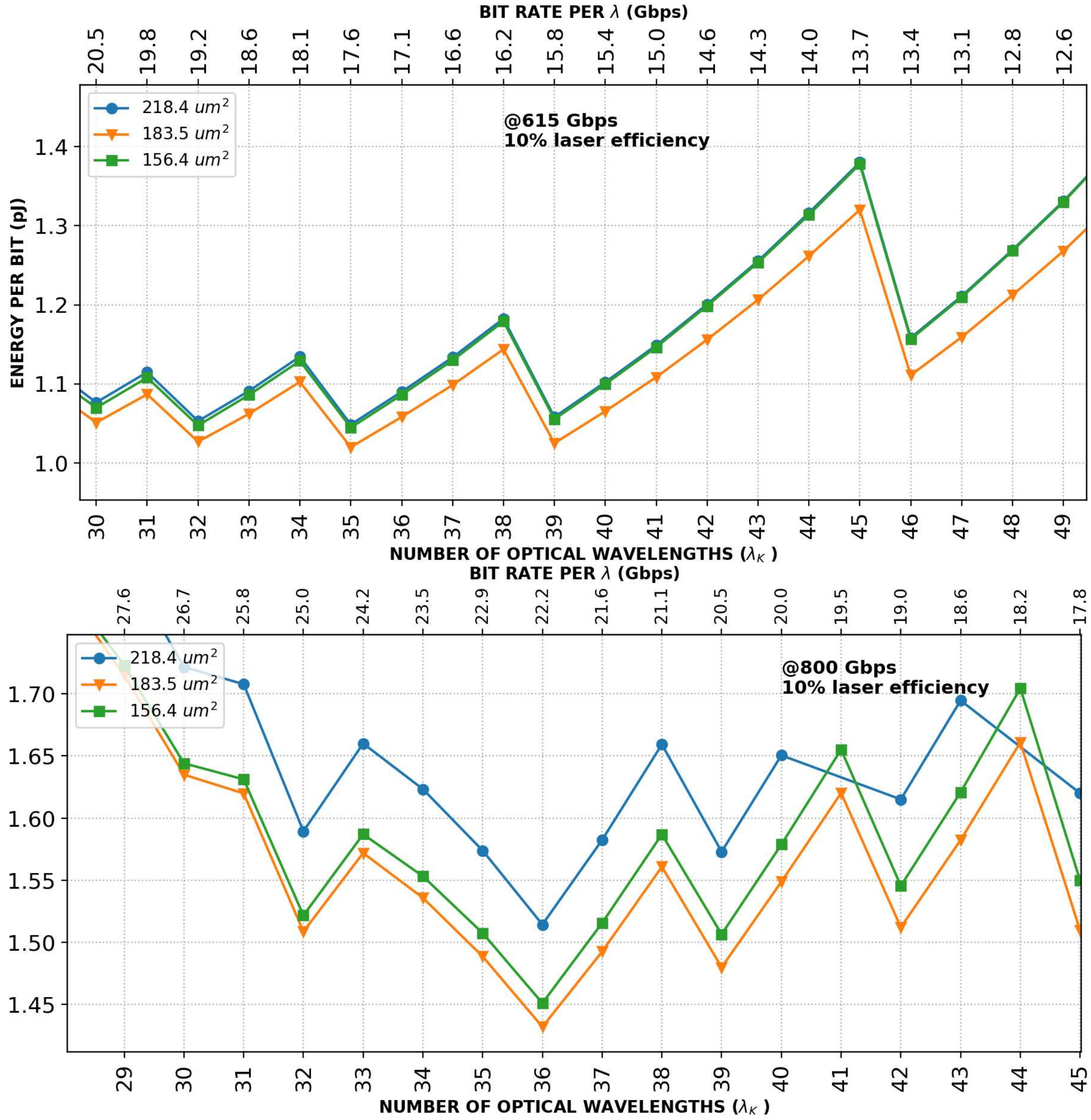


Figure 5.18: SiP link energy-per-bit using a laser with 10% efficiency. Top: at 615 Gbps bandwidth, Bottom: at 800 Gbps bandwidth.

From Figure 5.20, we make three observations: (i) with the current setup shown in Table 5.4, the energy per bit grows exponentially for aggregated bandwidths above 2500 Gbps and above the average DDR off-chip data movement energy at 3000 Gbps; (ii) the most energy-efficient number of wavelengths grows approximately linearly with the aggregated bandwidth; (iii) demonstrated fabrication feasibility is highlighted up to 800 Gbps, and the region to the right is estimated with PhoenixSim, yet currently not feasible. Accordingly, we can say that OCM must include new and more efficient optical device models to grow beyond the 3000 Gbps mark. Furthermore, this growth must include optimizing MRRs -or similar devices- to either multiply the number of possible wavelengths or raise the bitrate per wavelength. The physics of this growth will be addressed in the Scaling of optical devices section.

We conclude that a bidirectional SiP link, formed by two unidirectional links using current SiP devices, can fit the bandwidth requirements of commodity DDR4 DRAM modules. OCM incurs a low energy overhead of only 10.2% compared to a non-disaggregated

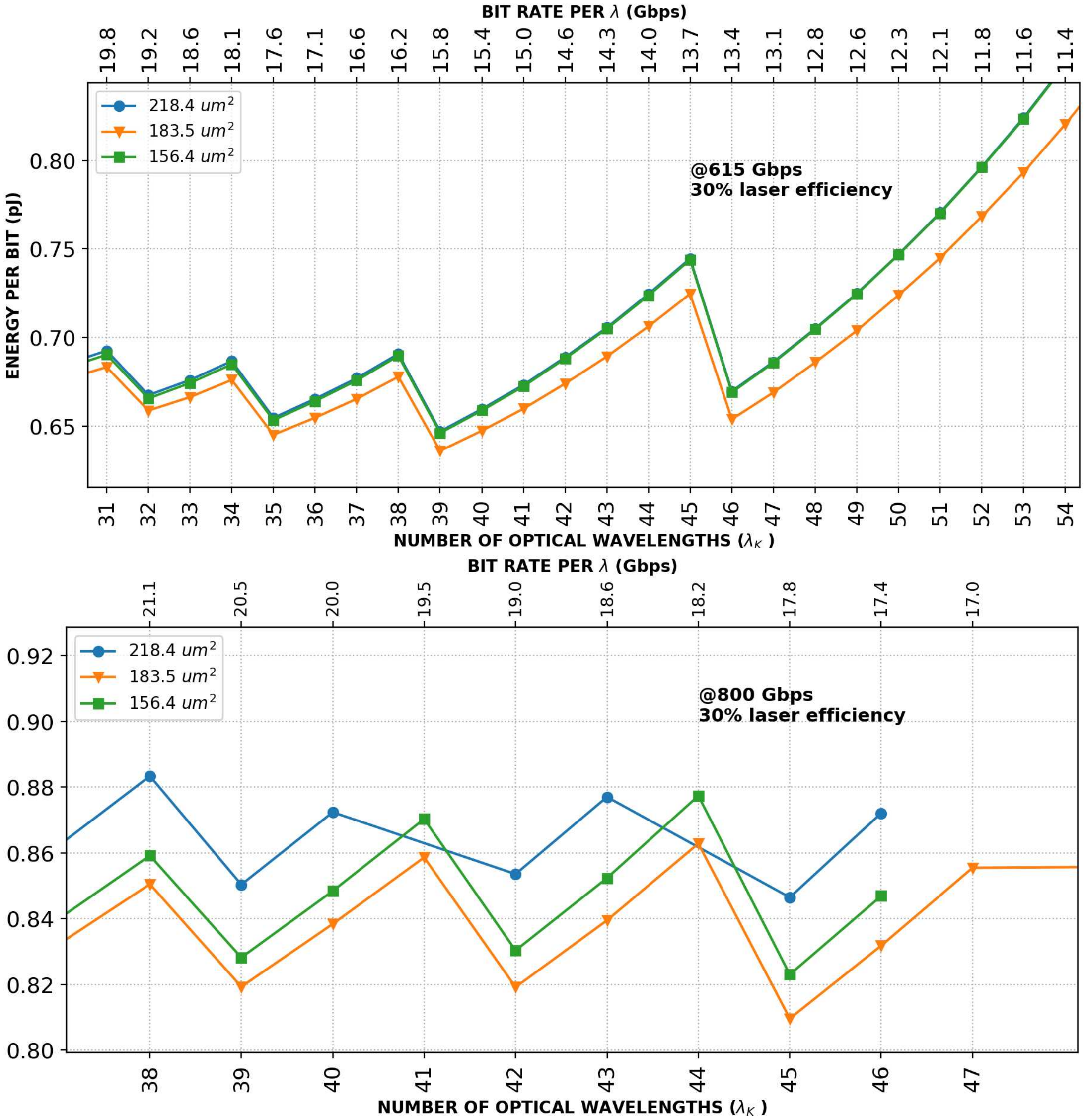


Figure 5.19: SiP link energy-per-bit using a laser with 30% efficiency. Top: at 615 Gbps bandwidth, Bottom: at 800 Gbps bandwidth.

DDR4 DRAM memory (the energy consumption of current DDR4 DRAM technology is ∼ 10pJ/bit [229]).

Scaling of optical devices. Silicon has an indirect energy bandgap in near infrared frequencies. Thus, active devices such as lasers or photodetectors cannot be fabricated using only a single material. Optical active devices on silicon are moving towards monolithic integration of other energy efficient materials. Namely, heterogeneous integration of III-V-group materials epitaxially grown on silicon [255], and the integration of two-dimensional [67] and one-dimensional [150, 73] materials to improve modulation, amplification, switching and photodetection. These techniques enable more efficient lasers, improve the sensitivity of photodetectors and reduce modulators driving power. These improvements can affect directly the SiP link estimated energy-per-bit, as shown in the 10% versus 30% integrated laser example discussed previously on this section.

Scaling of lasers on a silicon platform has advanced from epitaxially grown III-V quantum wells, in the scale of tens of nanometers, towards epitaxially grown quantum

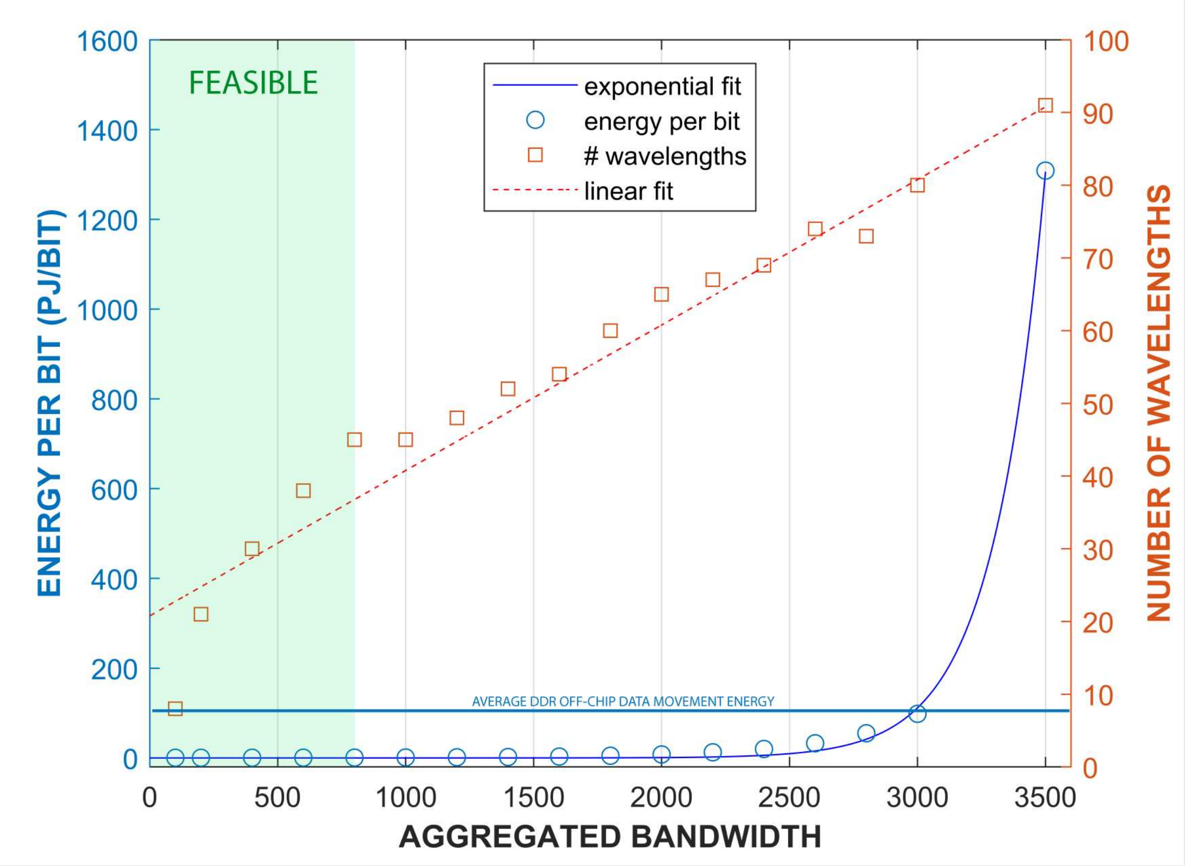


Figure 5.20: Minimum SiP link energy consumption and number of wavelengths tendency as a function of the aggregated bandwidth. Results that are currently feasible are highlighted. Aggregated bandwidth is measured in Gbps.

dots. Although the physics of transversal confinement of light does not change over the years, the cavity length of the device has been shrunk down and its efficiency has improved[73]. In this work we considered as feasible an epitaxially grown on silicon 10% and 30% quantum-well lasers.

Photodetectors require active materials with an direct energy bandgap in the infrared. Germanium has been widely used for this purpose. However, defending the tendency of new materials for smaller footprint, quantum dot photodetors with III-V materials, and the use of two-dimensional materials [154] are also relevant in the literature. In this work, we considered a Germanium optimized photodetector with a high sensitivity [81] which is feasible and CMOS compatible on a high scale.

Lastly silicon modulators are important features of the SiP link, and the ones that require the biggest footprint. Traditionally an electrooptic effect is induced on silicon by doping the MRR optical waveguide slightly [25, 24]. This is the modulation method we use in this work. However, as seen in Figure 5.20, although there is a predictable linear evolution of the required number of λ's, the energy-per-bit grows exponentially beyond the terahertz aggregated bandwidth, making it nonviable to use all configurations as they are presented in the future. However, it is enough to include new devices in the PhoenixSim platform to estimate a new path for the growth of SiP links in OCM. The reader should also note that works with hybrid Silicon photonics and 2D semiconductor monolayers were demonstrated [67].

5.5 Final Considerations

We propose and evaluate Optically Connected Memory (OCM), a new optical architecture for disaggregated main memory systems, compatible with current DDR DRAM technology. OCM uses a Silicon Photonics (SiP) platform that enables memory disaggregation with low energy-per-bit overhead. Our evaluation shows that, for the bandwidth required by current DDR standards, OCM has significantly better energy efficiency than conventional electrical NIC-based communication systems, and it incurs a low energy overhead of only 10.7% compared to DDR DRAM memory. Using system-level simulation to evaluate our OCM model on real applications, we find that OCM performs 5.5 times faster than a 40G NIC-based disaggregated memory. We conclude that OCM is a promising step towards future data centers with disaggregated main memory.

Chapter 6

Conclusions

In this thesis, we presented our study for photonics opportunities in modern computing systems. We identify three main contributions. First, we design and evaluate a new architecture with optical devices for the main memory system. A full-optical main memory system improves performance while allows rethinking the cache by reducing its size, while it imposes challenges on efficient control over the silicon photonics plane as massive memory cell fabrication matures. Second, we proposed an optical architecture for reconfigurable interconnection of multi-GPU devices. We demonstrated it delivers efficient data movement between multi-GPU memory devices. Third, we designed and evaluated extensively an optically connected memory for data center disaggregation. We demonstrated on both system-level and SiP link levels that optical disaggregation is viable to achieve energy-efficient Tbps data centers (≈1 pJ/bit). We studied how we can use photonics in computing systems with multicore processors and GPUs by: a) evaluating the system's performance, b) estimating the number of optical devices and their energy consumption using realistic SiP link models.

We identified two main challenges developing our studies. First, it is important to define realistic scenarios and parameters for the optical devices in a computing system. For example, while it is feasible to place optical transceivers close to the processing elements. A SiP link model based on real devices' measures helps define the number of optical devices according to the desired bandwidth. We overcome this challenge by collaborating with the Lightwave Research Laboratory from Columbia University, which focuses on experimental optical systems and develops the PhoenixSim model for SiP links. Second, laboratory experimental testbeds are envisioned as proof of concept. Then it is challenging to experiment with a complete system execution. We overcome this challenge by working with accurate system-level simulators to execute real workloads.

From the studies carried out in this thesis, we summarize our conclusions from each chapter:

- Chapter 3 introduces a full-optical main memory system. Silicon Photonics fabrics are building blocks for next-generation interconnects because of three main characteristics: a) high-bandwidth, b) CMOS affinity, and c) energy efficiency. To fully exploit the benefit of nanosecond switching, it is required but still not feasible: a) *ns* order setup time when a light path is established (including synchronization and clock recovery), b) an efficient control plane for massive switch activation providing adequate voltage levels. For a full-optical memory implementation, the massive integration of non-linear optics memory cells must reach maturity. Our evaluation shows that a full-optical memory allows us to reduce the cache levels with a latency in the order of ns, if control delay can be reduced from the curent μs order. We find that a full-optical memory system with a reduced 2 KB L1 data cache could perform $\approx 23\%$ slower than a conventional multicore system with L1 and L2 caches of 64KB.

- In Chapter 4, we proposed and evaluated an optically reconfigurable multi-GPU architecture that enables bandwidth reallocation between processing elements based on bandwidth-steering. Our architecture evaluation shows up to 20% data rate improvement with deep learning applications. We conclude that our static analysis shows that this improvement has a reasonable tradeoff of $\approx$ four additional receivers on the GPU side. For example, if a GPU has 4 sublinks, then it has 4 receivers to sustain communication. Then, we need to double the overall receiver area (add 4 additional receivers) on the GPU to benefit from a performance improvement. The extra receivers allow assigning extra links from other GPUs that are experimenting with sublink underutilization.

- In Chapter 5, we proposed and evaluated the Optically Connected Memory (OCM) architecture for main memory disaggregation. We designed and estimated the required number of photonic devices (i.e., micro-rings) to establish SiP links to sustain the required bandwidth of DDR4 DRAM technology. Our results show a low overhead of 10.7% energy-per-bit compared to DDR DRAM operation. Our performance evaluation, as expected, show a slowdown for rack distance disaggregation. However, we conclude that this slowdown can be considerably alleviated using a DDR4 based DRAM cache.

- We conclude that OCM and the optically reconfigurable multi-GPU architectures are initial steps towards future computing systems design with photonics. We can use our results to shape experimental testbeds for testing a specific usage scenario of the optical devices.

Publications

We published the following works with our contributions:

- **Gonzalez, J.**, Gazman, A., Hattink, M., Palma, M. G., Bahadori, M., Rubio-Noriega, R., Orosa, L., Mutlu, O., Bergman, K., Azevedo, R.. "Optically Connected Memory for Disaggregated Data Centers." 2020 IEEE 32nd International Symposium on Computer Architecture and High Performance Computing (SBAC-PAD). IEEE, 2020.

- Anderson, E., **González, J.**, Gazman, A., Azevedo, R., Bergman, K.. "Optically connected and reconfigurable GPU architecture for optimized peer-to-peer access." 2018 International Symposium on Memory Systems (MEMSYS). 2018.

- **Gonzalez, J.**, Orosa, L., Azevedo, R.. Architecting a computer with a full optical RAM. In 2016 IEEE International Conference on Electronics, Circuits and Systems (ICECS). IEEE. 2016.

We also have the following work under evaluation:

- **Gonzalez, J.**, Palma, M. G., Rubio-Noriega, R., Orosa, L., Mutlu, O., Bergman, K., Azevedo, R.. "Optically Connected Memory for Disaggregated Data Centers (extended version)" Journal of Parallel and Distributed Computing (JPDC). 2021 (*Submitted*).

Future work

- A dynamic analysis during workload execution can be studied for the Optically reconfigurable multi-GPU architecture. This type of analysis faces the main challenge of efficient control of the optical switch for bandwidth steering. It is feasible to define μs order observation windows for bandwidth steering estimation to define the control over switching, considering that memory communication in GPUs is in the same latency order. In addition, SiP link design can be analyzed to determine the constraints (e.g., energy-per-bit, number of SiP devices) to sustain multiple GPUs.

- A multiprocessor scenario can be evaluated for main memory disaggregation. Dynamic workload classification is required to determine subtilization and overprovision of the memory SiP links. For efficient optical switching, operating system aware mechanisms, such as page migration, can be studied to define the optical switch control.

- Experimental demonstration in a photonic testbed can be implemented for both multi-GPU and main memory disaggregation. As FPGAs can be used with photonic devices, the evaluation faces the following challenges: a) implement and execute representative benchmarks of data centers workloads, b) control the optical switches during execution based on the workload, and c) implement the required hardware modules to evaluate our architectures.

Bibliography

[1] Tensorflow Flowers Dataset. `download.tensorflow.org/example_images/flower_photos.tgz`. [Online; accessed 28-April-2019].

[2] JEDEC DDR4 Standard. `https://www.jedec.org/`, 2012.

[3] Martín Abadi, Paul Barham, Jianmin Chen, Zhifeng Chen, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Geoffrey Irving, Michael Isard, et al. Tensorflow: A system for large-scale machine learning. In *12th {USENIX} Symposium on Operating Systems Design and Implementation ({OSDI} 16)*, pages 265–283, 2016.

[4] Bulent Abali, Richard J Eickemeyer, Hubertus Franke, Chung-Sheng Li, and Marc A Taubenblatt. Disaggregated and optically interconnected memory: when will it be cost effective? *arXiv:1503.01416*, 2015.

[5] Nathan C Abrams, Qixiang Cheng, Madeleine Glick, Moises Jezzini, Padraic Morrissey, Peter O'Brien, and Keren Bergman. Silicon photonic 2.5 d multi-chip module transceiver for high-performance data centers. *Journal of Lightwave Technology*, 38(13):3346–3357, 2020.

[6] Philippe P Absil, Peter De Heyn, Hongtao Chen, Peter Verheyen, Guy Lepage, Marianna Pantouvaki, Jeroen De Coster, Amit Khanna, Youssef Drissi, Dries Van Thourhout, et al. Imec iSiPP25G silicon photonics: a robust CMOS-based photonics technology platform. In *Silicon Photonics X*, 2015.

[7] Mohammad Shahanshah Akhter, Paul Somogyi, Chen Sun, Mark Wade, Roy Meade, Pavan Bhargava, Sen Lin, and Nandish Mehta. Wavelight: A monolithic low latency silicon-photonics communication platform for the next-generation disaggregated cloud data centers. In *HOTI*, 2017.

[8] Nikolaos Alachiotis, Andreas Andronikakis, Orion Papadakis, Dimitris Theodoropoulos, Dionisios Pnevmatikatos, Dimitris Syrivelis, Andrea Reale, Kostas Katrinis, George Zervas, Vaibhawa Mishra, et al. dredbox: A disaggregated architectural perspective for data centers. In *Hardware Accelerators in Data Centers*, pages 35–56. Springer, 2019.

[9] Andrew Alduino and Mario Paniccia. Interconnects: Wiring electronics with light. *Nature Photonics*, 1(3):153, 2007.

[10] Theoni Alexoudi, George Theodore Kanellos, and Nikos Pleros. Optical ram and integrated optical memories: a survey. *Light: Science & Applications*, 9(1):1–16, 2020.

[11] Luca Alloatti, Robert Palmer, Sebastian Diebold, Kai Philipp Pahl, Baoquan Chen, Raluca Dinu, Maryse Fournier, Jean-Marc Fedeli, Thomas Zwick, Wolfgang Freude, et al. 100 ghz silicon–organic hybrid modulator. *Light: Science & Applications*, 3(5):e173–e173, 2014.

[12] Md Zahangir Alom, Tarek M Taha, Christopher Yakopcic, Stefan Westberg, Paheding Sidike, Mst Shamima Nasrin, Brian C Van Esesn, Abdul A S Awwal, and Vijayan K Asari. The history began from alexnet: A comprehensive survey on deep learning approaches. *arXiv preprint arXiv:1803.01164*, 2018.

[13] M. W. AlTaha, H. Jayatilleka, Z. Lu, J. F. Chung, D. Celo, D. Goodwill, E. Bernier, S. Mirabbasi, L. Chrostowski, and S. Shekhar. Monitoring and automatic tuning and stabilization of a 2&#x000d7;2 mzi optical switch for large-scale wdm switch networks. *Opt. Express*, 27(17):24747–24764, Aug 2019.

[14] AMD. Firepro directgma. `https://gpuopen.com/compute-product/direct-gma/`. [Online; accessed 22-February-2020].

[15] Erik Anderson, Jorge González, Alexander Gazman, Rodolfo Azevedo, and Keren Bergman. Optically connected and reconfigurable gpu architecture for optimized peer-to-peer access. In *Proceedings of the International Symposium on Memory Systems*, pages 257–258, 2018.

[16] Erik F Anderson, Alexander Gazman, Ziyi Zhu, Maarten Hattink, and Keren Bergman. Reconfigurable silicon photonic platform for memory scalability and disaggregation. In *OFC*, 2018.

[17] K. Ando, K. Ueyoshi, K. Orimo, H. Yonekawa, S. Sato, H. Nakahara, M. Ikebe, T. Asai, S. Takamaeda-Yamazaki, T. Kuroda, and M. Motomura. Brein memory: A 13-layer 4.2 k neuron/0.8 m synapse binary/ternary reconfigurable in-memory deep neural network accelerator in 65 nm cmos. In *2017 Symposium on VLSI Circuits*, pages C24–C25, June 2017.

[18] Aayush Ankit, Izzat El Hajj, Sai Rahul Chalamalasetti, Geoffrey Ndu, Martin Foltin, R Stanley Williams, Paolo Faraboschi, Wen-mei W Hwu, John Paul Strachan, Kaushik Roy, et al. Puma: A programmable ultra-efficient memristor-based accelerator for machine learning inference. In *Proceedings of the Twenty-Fourth International Conference on Architectural Support for Programming Languages and Operating Systems*, pages 715–731, 2019.

[19] Apostolos Argyris, Julián Bueno, and Ingo Fischer. Photonic machine learning implementation for signal recovery in optical communications. *Scientific Reports*, 8(1):8487, 2018.

[20] Krste Asanovic and David Patterson. Firebox: A hardware building block for 2020 warehouse-scale computers. In *USENIX FAST*, 2014.

[21] Amir H Atabaki, Sajjad Moazeni, Fabio Pavanello, Hayk Gevorgyan, Jelena Notaros, Luca Alloatti, Mark T Wade, Chen Sun, Seth A Kruger, Kenaish A Qubaisi, et al. Monolithic optical transceivers in 65 nm bulk cmos. In *2018 Optical Fiber Communications Conference and Exposition (OFC)*, pages 1–3. IEEE, 2018.

[22] Ammar Ahmad Awan, Hari Subramoni, and Dhabaleswar K Panda. An in-depth performance characterization of cpu-and gpu-based dnn training on modern architectures. In *Proceedings of the Machine Learning on HPC Environments*, pages 1–8. 2017.

[23] M. Bahadori, R. Polster, S. Rumley, Y. Thonnart, J. Gonzalez-Jimenez, and K. Bergman. Energy-bandwidth design exploration of silicon photonic interconnects in 65nm CMOS. In *OI*, 2016.

[24] Meisam Bahadori, Alexander Gazman, Natalie Janosik, Sébastien Rumley, Ziyi Zhu, Robert Polster, Qixiang Cheng, and Keren Bergman. Thermal rectification of integrated microheaters for microring resonators in silicon photonics platform. *Journal of Lightwave Technology*, pages 773–788, 2018.

[25] Meisam Bahadori, Mahdi Nikdast, Sébastien Rumley, Liang Yuan Dai, Natalie Janosik, Thomas Van Vaerenbergh, Alexander Gazman, Qixiang Cheng, Robert Polster, and Keren Bergman. Design space exploration of microring resonators in silicon photonic interconnects: impact of the ring curvature. *Journal of lightwave technology*, 36(13):2767–2782, 2018.

[26] Meisam Bahadori, Sébastien Rumley, Dessislava Nikolova, and Keren Bergman. Comprehensive design space exploration of silicon photonic interconnects. *Journal of Lightwave Technology*, 2016.

[27] Meisam Bahadori, Sébastien Rumley, Robert Polster, Alexander Gazman, Matt Traverso, Mark Webster, Kaushik Patel, and Keren Bergman. Energy-performance optimized design of silicon photonic interconnection networks for high-performance computing. In *DATE*, 2017.

[28] David H Bailey, Eric Barszcz, John T Barton, David S Browning, Robert L Carter, Leonardo Dagum, Rod A Fatoohi, Paul O Frederickson, Thomas A Lasinski, Rob S Schreiber, et al. The nas parallel benchmarks summary and preliminary results. In *Supercomputing'91: Proceedings of the 1991 ACM/IEEE conference on Supercomputing*, pages 158–165. IEEE, 1991.

[29] Janibul Bashir, Eldhose Peter, and Smruti R Sarangi. A survey of on-chip optical interconnects. *ACM Computing Surveys (CSUR)*, 51(6):1–34, 2019.

[30] Christopher Batten, Ajay Joshi, Jason Orcutt, Anatol Khilo, Benjamin Moss, Charles W Holzwarth, Miloš A Popovic, Hanqing Li, Henry I Smith, Judy L Hoyt,

et al. Building Many-core Processor-to-DRAM Networks with Monolithic CMOS Silicon Photonics. *MICRO*, 2009.

[31] Scott Beamer, Krste Asanovic, and David A. Patterson. The GAP Benchmark Suite. *CoRR*, 2015.

[32] Scott Beamer, Chen Sun, Yong-Jin Kwon, Ajay Joshi, Christopher Batten, Vladimir Stojanović, and Krste Asanović. Re-architecting dram memory systems with monolithically integrated silicon photonics. In *ISCA*, 2010.

[33] Tal Ben-Nun and Torsten Hoefler. Demystifying parallel and distributed deep learning: An in-depth concurrency analysis. *ACM Computing Surveys (CSUR)*, 52(4):1–43, 2019.

[34] Brad Benton. CCIX, Gen-Z, OpenCAPI: Overview & Comparison. In *OpenFabrics Workshop*, 2017.

[35] Keren Bergman, John Shalf, George Michelogiannakis, Sebastien Rumley, Larry Dennison, and Monia Ghobadi. Pine: An energy efficient flexibly interconnected photonic data center architecture for extreme scalability. In *OI*, 2018.

[36] Christian Bienia, Sanjeev Kumar, and Kai Li. PARSEC vs. SPLASH-2: A Quantitative Comparison of Two Multithreaded Benchmark Suites on Chip-Multiprocessors. In *IISWC*, 2008.

[37] Christian Bienia, Sanjeev Kumar, Jaswinder Pal Singh, and Kai Li. The PARSEC Benchmark Suite: Characterization and Architectural Implications. In *PACT*, 2008.

[38] Wim Bogaerts and Lukas Chrostowski. Silicon photonics circuit design: methods, tools and challenges. *Laser & Photonics Reviews*, 12(4):1700237, 2018.

[39] Maxence Bouvier, Alexandre Valentian, Thomas Mesquida, Francois Rummens, Marina Reyboz, Elisa Vianello, and Edith Beigne. Spiking neural networks hardware implementations and challenges: A survey. *J. Emerg. Technol. Comput. Syst.*, 15(2), April 2019.

[40] C. A. Brackett. Dense wavelength division multiplexing networks: principles and applications. *IEEE Journal on Selected Areas in Communications*, 1990.

[41] D. Brunina, D. Liu, and K. Bergman. An Energy-Efficient Optically Connected Memory Module for Hybrid Packet- and Circuit-Switched Optical Networks. *JSTQE*, 2013.

[42] Daniel Brunina, C Lai, A Garg, and Keren Bergman. Building Data Centers with Optically Connected Memory. *JOCN*, 2011.

[43] Daniel Brunina, Xiaoliang Zhu, Kishore Padmaraju, Long Chen, Michal Lipson, and Keren Bergman. 10-Gb/s WDM Optically-Connected Memory System Using Silicon Microring Modulators. In *ECOC*, 2012.

[44] James Bucek, Klaus-Dieter Lange, and Jóakim v. Kistowski. SPEC CPU2017: Next-generation Compute Benchmark. In *ICPE*, 2018.

[45] B. B. Buckley, S. T. M. Fryslie, K. Guinn, G. Morrison, A. Gazman, Y. Shen, K. Bergman, M. L. Mashanovitch, and L. A. Johansson. WDM source based on high-power, efficient 1280-nm DFB Lasers for Terabit Interconnect Technologies. *IEEE Photonics Technology Letters*, 2018.

[46] Brandon Buscaino, Mian Zhang, Marko Lončar, and Joseph M Kahn. Design of efficient resonator-enhanced electro-optic frequency comb generators. *Journal of Lightwave Technology*, 38(6):1400–1413, 2020.

[47] Maria Carla Calzarossa, Luisa Massari, and Daniele Tessera. Workload characterization: A survey revisited. *ACM Comput. Surv.*, 48(3), February 2016.

[48] Cristóbal Camarero, Enrique Vallejo, and Ramón Beivide. Topological characterization of hamming and dragonfly networks and its implications on routing. *ACM Trans. Archit. Code Optim.*, 11(4), December 2014.

[49] Henri Casanova, Arnaud Giersch, Arnaud Legrand, Martin Quinson, and Frédéric Suter. Versatile, scalable, and accurate simulation of distributed applications and platforms. *Journal of Parallel and Distributed Computing*, 74(10):2899–2917, June 2014.

[50] Kumar Chellapilla, Sidd Puri, and Patrice Simard. High Performance Convolutional Neural Networks for Document Processing. In Guy Lorette, editor, *Tenth International Workshop on Frontiers in Handwriting Recognition*, La Baule (France), October 2006. Université de Rennes 1, Suvisoft. http://www.suvisoft.com.

[51] C H Chen, S Matsuo, K Nozaki, A Shinya, and T Sato. All-optical memory based on injection-locking bistability in photonic crystal lasers. *Optics ...*, 2011.

[52] Chen Chen, Chongfu Zhang, Wei Zhang, Wei Jin, and Kun Qiu. Scalable and reconfigurable generation of flat optical comb for wdm-based next-generation broadband optical access networks. *Optics Communications*, 321:16–22, 2014.

[53] Kun-Chih (Jimmy) Chen, Masoumeh Ebrahimi, Ting-Yi Wang, and Yuch-Chi Yang. Noc-based dnn accelerator: A future design paradigm. In *Proceedings of the 13th IEEE/ACM International Symposium on Networks-on-Chip*, NOCS '19, New York, NY, USA, 2019. Association for Computing Machinery.

[54] Li Chen, Qiang Xu, Michael G Wood, and Ronald M Reano. Hybrid silicon and lithium niobate electro-optical ring modulator. *Optica*, 1(2):112–118, 2014.

[55] W. Chen, K. Ye, Y. Wang, G. Xu, and C. Xu. How Does the Workload Look Like in Production Cloud? Analysis and Clustering of Workloads on Alibaba Cluster Trace. In *ICPADS*, 2018.

[56] Xi Chen, John Wilson, John Poulton, Rizwan Bashirullah, and Tom Gray. High-speed low-power on-chip global signaling design overview. `https://research.nvidia.com/publication/2015-01_High-speed-Low-power-On-chip`. [Online; accessed 22-February-2020].

[57] Xia Chen, Christy KY Fung, Yi Min Chen, and Hon K Tsang. Subwavelength Waveguide Grating Coupler for Fiber-to-Chip Coupling on SOI with 80nm 1dB-bandwidth. In *CLEO*, 2011.

[58] Qixiang Cheng, Meisam Bahadori, Madeleine Glick, Sébastien Rumley, and Keren Bergman. Recent advances in optical technologies for data centers: a review. *Optica*, 5(11):1354–1370, 2018.

[59] Qixiang Cheng, Liang Yuan Dai, Nathan C Abrams, Yu-Han Hung, Padraic E Morrissey, Madeleine Glick, Peter O'Brien, and Keren Bergman. Ultralow-crosstalk, strictly non-blocking microring-based optical switch. *Photonics Research*, 7(2):155–161, 2019.

[60] Qixiang Cheng, Yishen Huang, Hao Yang, Meisam Bahadori, Nathan Abrams, Xiang Meng, Madeleine Glick, Yang Liu, Michael Hochberg, and Keren Bergman. Silicon photonic switch topologies and routing strategies for disaggregated data centers. *IEEE Journal of Selected Topics in Quantum Electronics*, 26(2):1–10, 2019.

[61] Qixiang Cheng, Sébastien Rumley, Meisam Bahadori, and Keren Bergman. Photonic switching in high performance datacenters. *Optics Express*, 2018.

[62] Zeshan Chishti and Berkin Akin. Memory system characterization of deep learning workloads. In *Proceedings of the International Symposium on Memory Systems*, MEMSYS '19, page 497–505, New York, NY, USA, 2019. Association for Computing Machinery.

[63] C. Chu, J. M. Hashmi, K. S. Khorassani, H. Subramoni, and D. K. Panda. High-performance adaptive mpi derived datatype communication for modern multi-gpu systems. In *2019 IEEE 26th International Conference on High Performance Computing, Data, and Analytics (HiPC)*, pages 267–276, Dec 2019.

[64] C. Chu, X. Lu, A. A. Awan, H. Subramoni, B. Elton, and D. K. Panda. Exploiting hardware multicast and gpudirect rdma for efficient broadcast. *IEEE Transactions on Parallel and Distributed Systems*, 30(3):575–588, March 2019.

[65] Dan Claudiu Cireşan, Ueli Meier, Luca Maria Gambardella, and Jürgen Schmidhuber. Deep, big, simple neural nets for handwritten digit recognition. *Neural Computation*, 22(12):3207–3220, 2010. PMID: 20858131.

[66] Milorad Cvijetic and Ivan Djordjevic. *Advanced optical communication systems and networks*. Artech House, 2013.

[67] Ipshita Datta, Sang Hoon Chae, Gaurang R Bhatt, Mohammad Amin Tadayon, Baichang Li, Yiling Yu, Chibeom Park, Jiwoong Park, Linyou Cao, DN Basov, et al. Low-loss composite photonic platform based on 2d semiconductor monolayers. *Nature Photonics*, 14(4):256–262, 2020.

[68] Lorenzo De Marinis, Marco Cococcioni, Piero Castoldi, and Nicola Andriolli. Photonic neural networks: A survey. *IEEE Access*, 7:175827–175841, 2019.

[69] Augustin Degomme, Arnaud Legrand, George S Markomanolis, Martin Quinson, Mark Stillwell, and Frédéric Suter. Simulating mpi applications: the smpi approach. *IEEE Transactions on Parallel and Distributed Systems*, 28(8):2387–2400, 2017.

[70] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009.

[71] S. Di, D. Kondo, and W. Cirne. Characterization and Comparison of Cloud versus Grid Workloads. In *CLOUD*, 2012.

[72] Jack Dongarra. Report on the fujitsu fugaku system. *University of Tennessee, Innovative Computing Laboratory*, page 7, 2020.

[73] Jianan Duan, Heming Huang, Bozhang Dong, Daehwan Jung, Justin C Norman, John E Bowers, and Frederic Grillot. 1.3-um reflection insensitive inas/gaas quantum dot lasers directly grown on silicon. *IEEE Photonics Technology Letters*, 31(5):345–348, 2019.

[74] Javier Duarte, Philip Harris, Scott Hauck, Burt Holzman, Shih-Chieh Hsu, Sergo Jindariani, Suffian Khan, Benjamin Kreis, Brian Lee, Mia Liu, et al. Fpga-accelerated machine learning inference as a service for particle physics computing. *Computing and Software for Big Science*, 3(1):13, 2019.

[75] Nicolas Dupuis, Fuad Doany, Russell A Budd, Laurent Schares, Christian W Baks, Daniel M Kuchta, Takako Hirokawa, and Benjamin G Lee. A 4× 4 electrooptic silicon photonic switch fabric with net neutral insertion loss. *Journal of Lightwave Technology*, 38(2):178–184, 2020.

[76] F. Eltes, J. Barreto, D. Caimi, S. Karg, A. A. Gentile, A. Hart, P. Stark, N. Meier, M. G. Thompson, J. Fompeyrine, and S. Abel. First cryogenic electro-optic switch on silicon with high bandwidth and low power tunability. In *2018 IEEE International Electron Devices Meeting (IEDM)*, pages 23.1.1–23.1.4, 2018.

[77] Felix Eltes, Gerardo E Villarreal-Garcia, Daniele Caimi, Heinz Siegwart, Antonio A Gentile, Andy Hart, Pascal Stark, Graham D Marshall, Mark G Thompson, Jorge Barreto, et al. An integrated optical modulator operating at cryogenic temperatures. *Nature Materials*, 19(11):1164–1168, 2020.

[78] Alexandros Emboras, Jens Niegemann, Ping Ma, Christian Haffner, Andreas Pedersen, Mathieu Luisier, Christian Hafner, Thomas Schimmel, and Juerg Leuthold. Atomic scale plasmonic switch. *Nano Letters*, 16(1):709–714, 2016.

[79] Alexandros Emboras, Jens Niegemann, Ping Ma, Christian Haffner, Andreas Pedersen, Mathieu Luisier, Christian Hafner, Thomas Schimmel, and Juerg Leuthold. Atomic Scale Plasmonic Switch. *NL*, 2016.

[80] Nicholas M Fahrenkopf, Colin McDonough, Gerald L Leake, Zhan Su, Erman Timurdogan, and Douglas D Coolbaugh. The aim photonics mpw: A highly accessible cutting edge technology for rapid prototyping of photonic integrated circuits. *IEEE Journal of Selected Topics in Quantum Electronics*, 25(5):1–6, 2019.

[81] Monireh Moayedi Pour Fard, Glenn Cowan, and Odile Liboiron-Ladouceur. Responsivity optimization of a high-speed germanium-on-silicon photodetector. *Optics express*, 24(24):27738–27752, 2016.

[82] Marjan Fariborz, Xian Xiao, Pouya Fotouhi, Roberto Proietti, and SJ Ben Yoo. Silicon photonic flex-lions for reconfigurable multi-gpu systems. *Journal of Lightwave Technology*, 39(4):1212–1220, 2021.

[83] Chenghao Feng, Zhoufeng Ying, Zheng Zhao, Rohan Mital, David Z Pan, and Ray T Chen. Analysis of microresonator-based logic gate for high-speed optical computing in integrated photonics. *IEEE Journal of Selected Topics in Quantum Electronics*, 26(2):1–8, 2019.

[84] David A Ferrucci. Introduction to “this is watson”. *IBM Journal of Research and Development*, 56(3.4):1–1, 2012.

[85] D Fitsios, T Alexoudi, A Bazin, P Monnier, R Raj, A Miliou, G T Kanellos, N Pleros, and F Raineri. Ultra-compact III–V-on-Si photonic crystal memory for flip-flop operation at 5 Gb/s. *Optics Express*, 2016.

[86] D Fitsios, K Vyrsokinos, A Miliou, and N Pleros. Memory Speed Analysis of Optical RAM and Optical Flip-Flop Circuits Based on Coupled SOA-MZI Gates. *IEEE Journal of Selected Topics in Quantum Electronics*, 2012.

[87] Denis Foley and John Danskin. Ultra-performance pascal gpu and nvlink interconnect. *IEEE Micro*, 37(2):7–17, 2017.

[88] Jeremy Fowers, Kalin Ovtcharov, Michael Papamichael, Todd Massengill, Ming Liu, Daniel Lo, Shlomi Alkalay, Michael Haselman, Logan Adams, Mahdi Ghandi, et al. A configurable cloud-scale dnn processor for real-time ai. In *2018 ACM/IEEE 45th Annual International Symposium on Computer Architecture (ISCA)*, pages 1–14. IEEE, 2018.

[89] Luiz M Franca-Neto. Field-programmable deep neural network (dnn) learning and inference accelerator: a concept. *arXiv preprint arXiv:1802.04899*, 2018.

[90] Alexander L Gaeta, Michal Lipson, and Tobias J Kippenberg. Photonic-chip-based frequency combs. *nature photonics*, 13(3):158–169, 2019.

[91] Debashis Ganguly, Ziyu Zhang, Jun Yang, and Rami Melhem. Interplay between hardware prefetcher and page eviction policy in cpu-gpu unified virtual memory. In *Proceedings of the 46th International Symposium on Computer Architecture*, pages 224–235, 2019.

[92] Ken Giewont, Karen Nummy, Frederick A Anderson, Javier Ayala, Tymon Barwicz, Yusheng Bian, Kevin K Dezfulian, Douglas M Gill, Thomas Houghton, Shuren Hu, et al. 300-mm monolithic silicon photonics foundry technology. *IEEE Journal of Selected Topics in Quantum Electronics*, 25(5):1–11, 2019.

[93] Madeleine Glick, Nathan C Abrams, Qixiang Cheng, Min Yee Teh, Yu-Han Hung, Oscar Jimenez, Songtao Liu, Yoshitomo Okawachi, Xiang Meng, Leif Johansson, et al. Pine: photonic integrated networked energy efficient datacenters (enlitened program). *IEEE/OSA Journal of Optical Communications and Networking*, 12(12):443–456, 2020.

[94] Madeleine Glick, Lionel C. Kimmerling, and Robert C. Pfahl. A Roadmap for Integrated Photonics. *OPN*, 2018.

[95] Jorge Gonzalez, Alexander Gazman, Maarten Hattink, Mauricio G Palma, Meisam Bahadori, Ruth Rubio-Noriega, Lois Orosa, Madeleine Glick, Onur Mutlu, Keren Bergman, et al. Optically connected memory for disaggregated data centers. In *2020 IEEE 32nd International Symposium on Computer Architecture and High Performance Computing (SBAC-PAD)*, pages 43–50. IEEE, 2020.

[96] Jorge Gonzalez, Lois Orosa, and Rodolfo Azevedo. Architecting a computer with a full optical ram. In *2016 IEEE International Conference on Electronics, Circuits and Systems (ICECS)*, pages 716–719. IEEE, 2016.

[97] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Advances in neural information processing systems*, pages 2672–2680, 2014.

[98] F Grillot, L Vivien, S Laval, D Pascal, and E Cassan. Size Influence on the Propagation Loss Induced by Sidewall Roughness in Ultrasmall SOI Waveguides. *PTL*, 2004.

[99] Juncheng Gu, Youngmoon Lee, Yiwen Zhang, Mosharaf Chowdhury, and Kang G Shin. Efficient memory disaggregation with infiniswap. In *NSDI*, 2017.

[100] Kaiyuan Guo, Shulin Zeng, Jincheng Yu, Yu Wang, and Huazhong Yang. [dl] a survey of fpga-based neural network inference accelerators. *ACM Trans. Reconfigurable Technol. Syst.*, 12(1), March 2019.

[101] Pengxing Guo, Weigang Hou, Lei Guo, Wei Sun, Chuang Liu, Hainan Bao, Luan HK Duong, and Weichen Liu. Fault-tolerant routing mechanism in 3d optical network-on-chip based on node reuse. *IEEE Transactions on Parallel and Distributed Systems*, 31(3):547–564, 2019.

[102] Ramyad Hadidi, Bahar Asgari, Burhan Ahmad Mudassar, Saibal Mukhopadhyay, Sudhakar Yalamanchili, and Hyesoon Kim. Demystifying the characteristics of 3D-stacked memories: A case study for hybrid memory cube. In *IISWC*, 2017.

[103] Christian Haffner, Daniel Chelladurai, Yuriy Fedoryshyn, Arne Josten, Benedikt Baeuerle, Wolfgang Heni, Tatsuhiko Watanabe, Tong Cui, Bojun Cheng, Soham Saha, et al. Low-loss plasmon-assisted electro-optic modulator. *Nature*, 556(7702):483–486, 2018.

[104] Song Han and William J Dally. Bandwidth-efficient deep learning. In *2018 55th ACM/ESDA/IEEE Design Automation Conference (DAC)*, pages 1–6. IEEE, 2018.

[105] M. Hattink, G. Di Guglielmo, L. P. Carloni, and K. Bergman. A scalable architecture for cnn accelerators leveraging high-performance memories. In *2020 IEEE High Performance Extreme Computing Conference (HPEC)*, pages 1–6, 2020.

[106] K. Hazelwood, S. Bird, D. Brooks, S. Chintala, U. Diril, D. Dzhulgakov, M. Fawzy, B. Jia, Y. Jia, A. Kalro, J. Law, K. Lee, J. Lu, P. Noordhuis, M. Smelyanskiy, L. Xiong, and X. Wang. Applied machine learning at facebook: A datacenter infrastructure perspective. In *2018 IEEE International Symposium on High Performance Computer Architecture (HPCA)*, pages 620–629, Feb 2018.

[107] John L Henning. SPEC CPU2006 Benchmark Descriptions. *ACM SIGARCH Computer Architecture News*, 2006.

[108] Tayler Hicklin Hetherington, Maria Lubeznov, Deval Shah, and Tor M Aamodt. Edge: Event-driven gpu execution. In *2019 28th International Conference on Parallel Architectures and Compilation Techniques (PACT)*, pages 337–353. IEEE, 2019.

[109] Martin T Hill, H de Waardt, G D Khoe, and H J S Dorren. Fast optical flip-flop by use of Mach-Zehnder interferometers. *Microwave and Optical Technology Letters*, 2001.

[110] Martin T Hill, Harmen J S Dorren, Tjibbe de Vries, Xaveer J M Leijtens, Jan Hendrik den Besten, Barry Smalbrugge, Yok-Siang Oei, Hans Binsma, Giok-Djan Khoe, and Meint K Smit. A fast low-power optical memory based on coupled micro-ring lasers. 2004.

[111] Gao Huang, Shichen Liu, Laurens Van der Maaten, and Kilian Q Weinberger. Condensenet: An efficient densenet using learned group convolutions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2752–2761, 2018.

[112] Yishen Huang, Qixiang Cheng, Yu-Han Hung, Hang Guan, Xiang Meng, Ari Novack, Matthew Streshinsky, Michael Hochberg, and Keren Bergman. Multi-stage 8× 8 silicon photonic switch based on dual-microring switching elements. *Journal of Lightwave Technology*, 2019.

[113] Yishen Huang, Qixiang Cheng, Anthony Rizzo, and Keren Bergman. Push—pull microring-assisted space-and-wavelength selective switch. *Opt. Lett.*, 45(10):2696–2699, May 2020.

[114] Nikhil Jain, Abhinav Bhatele, Louis H Howell, David Böhme, Ian Karlin, Edgar A León, Misbah Mubarak, Noah Wolfe, Todd Gamblin, and Matthew L Leininger. Predicting the performance impact of different fat-tree configurations. In *Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis*, pages 1–13, 2017.

[115] N. Janosik, Q. Cheng, M. Glick, Y. Huang, and K. Bergman. High-resolution silicon microring based architecture for optical matrix multiplication. In *2019 Conference on Lasers and Electro-Optics (CLEO)*, pages 1–2, May 2019.

[116] J. Jiang, D. J. Goodwill, P. Dumais, D. Celo, C. Zhang, H. Mehrvar, M. Rad, E. Bernier, M. Li, F. Zhao, C. Zhang, J. He, Y. Ding, Y. Wei, W. Liu, X. Tu, and D. Geng. 16x16 silicon photonic switch with nanosecond switch time and low-crosstalk architecture. In *45th European Conference on Optical Communication (ECOC 2019)*, pages 1–4, 2019.

[117] X. Jiang, N. Madan, L. Zhao, M. Upton, R. Iyer, S. Makineni, D. Newell, Y. Solihin, and R. Balasubramonian. CHOP: Adaptive Filter-Based DRAM Caching for CMP Server Platforms. In *HPCA*, 2010.

[118] Norman P Jouppi, Cliff Young, Nishant Patil, David Patterson, Gaurav Agrawal, Raminder Bajwa, Sarah Bates, Suresh Bhatia, Nan Boden, Al Borchers, et al. In-datacenter performance analysis of a tensor processing unit. In *Proceedings of the 44th Annual International Symposium on Computer Architecture*, pages 1–12, 2017.

[119] G T Kanellos, T Alexoudi, D Fitsios, C Vagionas, P Maniotis, S Papaioannou, A Miliou, and N Pleros. WDM-enabled optical RAM architectures for ultra-fast, low-power optical cache memories. In *Transparent Optical Networks (ICTON), 2013 15th International Conference on*, 2013.

[120] Kaan Kara, Dan Alistarh, Gustavo Alonso, Onur Mutlu, and Ce Zhang. Fpga-accelerated dense linear machine learning: A precision-convergence trade-off. In *2017 IEEE 25th Annual International Symposium on Field-Programmable Custom Computing Machines (FCCM)*, pages 160–167. IEEE, 2017.

[121] Kostas Katrinis, Dimitris Syrivelis, D Pnevmatikatos, Georgios Zervas, Dimitris Theodoropoulos, Iordanis Koutsopoulos, K Hasharoni, Daniel Raho, Christian Pinto, F Espina, et al. Rack-scale disaggregated cloud data centers: The dReDBox project vision. *DATE*, 2016.

[122] Hitoshi Kawaguchi. Ultrafast all-optical memory operation using a polarization bistable VCSEL. In *2010 12th International Conference on Transparent Optical Networks (ICTON)*, 2010.

[123] Kimberly Keeton. The machine: An architecture for memory-centric computing. In *Workshop on Runtime and Operating Systems for Supercomputers (ROSS)*, 2015.

[124] Shahram Keyvaninia, Muhammad Muneeb, Stevan Stanković, PJ Van Veldhoven, Dries Van Thourhout, and Günther Roelkens. Ultra-thin dvs-bcb adhesive bonding of iii-v wafers, dies and multiple dies to a patterned silicon-on-insulator substrate. *Optical Materials Express*, 3(1):35–46, 2013.

[125] Jacob B Khurgin. How to deal with the loss in plasmonics and metamaterials. *Nature Publishing Group*, 2015.

[126] G. Kim, J. Kim, J. H. Ahn, and J. Kim. Memory-centric system interconnect design with hybrid memory cubes. In *PACT*, 2013.

[127] David Kirk et al. Nvidia cuda software and gpu parallel computing architecture. In *ISMM*, volume 7, pages 103–104, 2007.

[128] Marouan Kouissi, Benoit Charbonnier, and Catherine Algani. Fast 2x2 mach-zehnder switch for optical interconnect applications. In *OSA Advanced Photonics Congress (AP) 2020 (IPR, NP, NOMA, Networks, PVLED, PSC, SPPCom, SOF)*, page PsTh2F.5. Optical Society of America, 2020.

[129] Ashok V Krishnamoorthy, JE Ford, KW Goossen, JA Walker, B Tseng, SP Hui, JE Cunningham, WY Jan, TK Woodward, MC Nuss, et al. The amoeba chip: an optoelectronic switch for multiprocessor networking using dense-wdm. In *Proceedings of Massively Parallel Processing Using Optical Interconnections*, pages 94–100. IEEE, 1996.

[130] Alex Krizhevsky, Vinod Nair, and Geoffrey Hinton. The cifar-10 dataset. *online: http://www. cs. toronto. edu/kriz/cifar. html*, 55, 2014.

[131] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012.

[132] Eiichi Kuramoch, Kengo Nozaki, Akihiko Shinya, Koji Takeda, Tomonari Sato, Shinji Matsuo, Hideaki Taniyama, Hisashi Sumikura, and Masaya Notomi. Large-scale integration of wavelength-addressable all-optical memories on a photonic crystal chip. *Nature Photonics*, 2014.

[133] Hyoukjun Kwon, Prasanth Chatarasi, Michael Pellauer, Angshuman Parashar, Vivek Sarkar, and Tushar Krishna. Understanding reuse, performance, and hardware cost of dnn dataflow: A data-centric approach. In *Proceedings of the 52nd Annual IEEE/ACM International Symposium on Microarchitecture*, MICRO '52, page 754–768, New York, NY, USA, 2019. Association for Computing Machinery.

[134] Yann LeCun. 1.1 deep learning hardware: Past, present, and future. In *2019 IEEE International Solid-State Circuits Conference-(ISSCC)*, pages 12–19. IEEE, 2019.

[135] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. *Nature*, 521(7553):436–444, 2015.

[136] Benjamin C Lee, Engin Ipek, Onur Mutlu, and Doug Burger. Architecting phase change memory as a scalable dram alternative. In *Proceedings of the 36th annual international symposium on Computer architecture*, pages 2–13, 2009.

[137] J. Lee, C. Kim, S. Kang, D. Shin, S. Kim, and H. Yoo. Unpu: A 50.6tops/w unified deep neural network accelerator with 1b-to-16b fully-variable weight bit-precision. In *2018 IEEE International Solid - State Circuits Conference - (ISSCC)*, pages 218–220, Feb 2018.

[138] Sergey Legtchenko, Hugh Williams, Kaveh Razavi, Austin Donnelly, Richard Black, Andrew Douglas, Nathanaël Cheriere, Daniel Fryer, Kai Mast, Angela Demke Brown, et al. Understanding rack-scale disaggregated storage. In *USENIX HotStorage 17*, 2017.

[139] Ang Li, Shuaiwen Leon Song, Jieyang Chen, Jiajia Li, Xu Liu, Nathan R Tallent, and Kevin J Barker. Evaluating modern gpu interconnect: Pcie, nvlink, nv-sli, nvswitch and gpudirect. *IEEE Transactions on Parallel and Distributed Systems*, 31(1):94–110, 2019.

[140] HH Li. Refractive index of silicon and germanium and its wavelength and temperature derivatives. *Journal of Physical and Chemical Reference Data*, 9(3):561–658, 1980.

[141] Ke Li, Shenghao Liu, David J. Thomson, Weiwei Zhang, Xingzhao Yan, Fanfan Meng, Callum G. Littlejohns, Han Du, Mehdi Banakar, Martin Ebert, Wei Cao, Dehn Tran, Bigeng Chen, Abdul Shakoor, Periklis Petropoulos, and Graham T. Reed. Electronic–photonic convergence for silicon photonics transmitters beyond 100 gbps on–off keying. *Optica*, 7(11):1514–1516, Nov 2020.

[142] Ming Li, Lin Zhang, Li-Min Tong, and Dao-Xin Dai. Hybrid silicon nonlinear photonics. *Photonics Research*, 6(5):B13–B22, 2018.

[143] Mu Li, David G Andersen, Jun Woo Park, Alexander J Smola, Amr Ahmed, Vanja Josifovski, James Long, Eugene J Shekita, and Bor-Yiing Su. Scaling distributed machine learning with the parameter server. In *11th {USENIX} Symposium on Operating Systems Design and Implementation ({OSDI} 14)*, pages 583–598, 2014.

[144] Mu Li, Li Zhou, Zichao Yang, Aaron Li, Fei Xia, David G Andersen, and Alexander Smola. Parameter server for distributed machine learning. In *Big Learning NIPS Workshop*, volume 6, page 2, 2013.

[145] Y. Li, S. Ghose, J. Choi, J. Sun, H. Wang, and O. Mutlu. Utility-Based Hybrid Memory Management. In *CLUSTER*, 2017.

[146] Kevin Lim, Jichuan Chang, Trevor Mudge, Parthasarathy Ranganathan, Steven K Reinhardt, and Thomas F Wenisch. Disaggregated memory for expansion and sharing in blade servers. In *ACM SIGARCH computer architecture news*, volume 37, pages 267–278. ACM, 2009.

[147] Yujun Lin, Song Han, Huizi Mao, Yu Wang, and William J Dally. Deep gradient compression: Reducing the communication bandwidth for distributed training. *arXiv preprint arXiv:1712.01887*, 2017.

[148] Jiuxing Liu, Balasubramanian Chandrasekaran, Weikuan Yu, Jiesheng Wu, Darius Buntinas, Sushmitha Kini, Dhabaleswar K Panda, and Pete Wyckoff. Microbenchmark performance comparison of high-speed cluster interconnects. *IEEE Micro*, 24(1):42–51, 2004.

[149] Liu Liu, Rajesh Kumar, Koen Huybrechts, Thijs Spuesens, Gunther Roelkens, Erik-Jan Geluk, Tjibbe de Vries, Philippe Regreny, Dries Van Thourhout, Roel Baets, and Geert Morthier. An ultra-small, low-power, all-optical flip-flop memory on a silicon chip. 2010.

[150] Songtao Liu, Xinru Wu, Daehwan Jung, Justin C Norman, MJ Kennedy, Hon K Tsang, Arthur C Gossard, and John E Bowers. High-channel-count 20 ghz passively mode-locked quantum dot laser directly grown on si with 4.1 tbit/s transmission capacity. *Optica*, 6(2):128–134, 2019.

[151] Y Liu, M T Hill, N Calabretta, H de Waardt, G D Khoe, and H J S Dorren. Three-state all-optical memory based on coupled ring lasers. *IEEE Photonics Technology Letters*, 2001.

[152] Y Liu, R McDougall, M T Hill, G Maxwell, S Zhang, R Harmon, F M Huijskens, L Rivers, H J S Dorren, and A Poustie. Packaged and hybrid integrated all-optical flip-flop memory. *Electronics Letters*, 2006.

[153] Scott Lloyd and Maya Gokhale. In-Memory Data Rearrangement for Irregular, Data-Intensive Computing. *Computer*, 2015.

[154] Mingsheng Long, Peng Wang, Hehai Fang, and Weida Hu. Progress, challenges, and opportunities for 2d material based photodetectors. *Advanced Functional Materials*, 29(19):1803807, 2019.

[155] Yun Long, Xueyuan She, and Saibal Mukhopadhyay. Design of reliable dnn accelerator with un-reliable reram. In *2019 Design, Automation & Test in Europe Conference & Exhibition (DATE)*, pages 1769–1774. IEEE, 2019.

[156] Lars Lundberg, Magnus Karlsson, Abel Lorences-Riesgo, Mikael Mazur, Jochen Schröder, Peter A Andrekson, et al. Frequency comb-based wdm transmission systems enabling joint signal processing. *Applied Sciences*, 8(5):718, 2018.

[157] H. Luo, T. Shahroodi, H. Hassan, M. Patel, A. G. Yağlıkçı, L. Orosa, J. Park, and O. Mutlu. CLR-DRAM: A Low-Cost DRAM Architecture Enabling Dynamic Capacity-Latency Trade-Off. In *ISCA*, 2020.

[158] P Maniotis, D Fitsios, G T Kanellos, and N Pleros. Optical Buffering for Chip Multiprocessors: A 16GHz Optical Cache Memory Architecture. *Lightwave Technology, Journal of*, 2013.

[159] P Maniotis, D Fitsios, GT Kanellos, and N Pleros. Optical buffering for chip multiprocessors: a 16ghz optical cache memory architecture. *Journal of lightwave technology*, 31(24):4175–4191, 2013.

[160] Pavlos Maniotis, Nicolas Dupuis, Laurent Schares, Daniel M. Kuchta, Marc A. Taubenblatt, and Benjamin G. Lee. Intra-node high-performance computing network architecture with nanosecond-scale photonic switches (invited). *J. Opt. Commun. Netw.*, 12(12):367–377, Dec 2020.

[161] Pavlos Maniotis, Savvas Gitzenis, Leandros Tassiulas, and Nikos Pleros. An optically-enabled chip–multiprocessor architecture using a single-level shared optical cache memory. *Optical Switching and Networking*, 22:54–68, 2016.

[162] Pablo Marin-Palomo, Juned N Kemal, Maxim Karpov, Arne Kordts, Joerg Pfeifle, Martin HP Pfeiffer, Philipp Trocha, Stefan Wolf, Victor Brasch, Miles H Anderson, et al. Microresonator-based solitons for massively parallel coherent optical communications. *Nature*, 546(7657):274–279, 2017.

[163] Mario Donato Marino. Architectural impacts of rfiop: Rf to address i/o pad and memory controller scalability. *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, 2018.

[164] John Markoff. Computer wins on 'jeopardy!': trivial, it's not. *New York Times*, 16, 2011.

[165] Goran Z Mashanovich. Optical switches and modulators in deep freeze. *Nature Materials*, 19(11):1135–1136, 2020.

[166] John D. McCalpin. Memory bandwidth and machine balance in current high performance computers. *IEEE Computer Society Technical Committee on Computer Architecture (TCCA) Newsletter*, pages 19–25, December 1995.

[167] Mellanox Technologies. FRONTERA: A new nsf-funded petascale computing system. `https://www.tacc.utexas.edu/systems/frontera`. [Online; accessed 22-February-2020].

[168] Mellanox Technologies. Introducing 200G HDR InfiniBand Solutions. `https://www.mellanox.com/related-docs/whitepapers/WP_Introducing_200G_HDR_InfiniBand_Solutions.pdf`. [Online; accessed 22-February-2020].

[169] William Maxwell Mellette, Glenn M Schuster, George Porter, George Papen, and Joseph E Ford. A scalable, partially configurable optical switch for data center networks. *Journal of Lightwave Technology*, 35(2):136–144, 2016.

[170] J. Meza, J. Chang, H. Yoon, O. Mutlu, and P. Ranganathan. Enabling Efficient and Scalable Hybrid Memories Using Fine-Granularity DRAM Cache Management. *CAL*, 2012.

[171] J. Meza, Q. Wu, S. Kumar, and O. Mutlu. Revisiting Memory Errors in Large-Scale Production Data Centers: Analysis and Modeling of New Trends from the Field. In *DSN*, 2015.

[172] Justin Meza, Yixin Luo, Samira Khan, Jishen Zhao, Yuan Xie, and Onur Mutlu. A Case for Efficient Hardware/Software Cooperative Management of Storage and Memory. *WEED*, 2013.

[173] David AB Miller. Optical interconnects to electronic chips. *Applied optics*, 2010.

[174] David AB Miller. Silicon photonics: Meshing optics with applications. *Nature Photonics*, 11(7):403–404, 2017.

[175] Kevin J Miller, Richard F Haglund, and Sharon M Weiss. Optical phase change materials in integrated silicon photonic devices. *Optical Materials Express*, 8(8):2415–2429, 2018.

[176] Mario Miscuglio and Volker J Sorger. Photonic tensor cores for machine learning. *arXiv*, pages arXiv–2002, 2020.

[177] V. Mishra, J. L. Benjamin, and G. Zervas. Demonstrating optically interconnected remote serial and parallel memory in disaggregated data centers. In *2020 Optical Fiber Communications Conference and Exhibition (OFC)*, pages 1–3, 2020.

[178] Takashi Mori, Yasuhiro Yamayoshi, and Hitoshi Kawaguchi. Low-switching-energy and high-repetition-frequency all-optical flip-flop operations of a polarization bistable vertical-cavity surface-emitting laser. *Applied Physics Letters*, 2006.

[179] Alvaro Moscoso-Mártir, Ali Tabatabaei-Mashayekh, Juliana Müller, Jovana Nojić, Rony Setter, Mad Nielsen, Anna Sandomirsky, Sylvie Rockman, Elad Mentovich, Florian Merget, et al. 8-channel wdm silicon photonics transceiver with soa and semiconductor mode-locked laser. *Optics express*, 26(19):25446–25459, 2018.

[180] Biswanath Mukherjee. Wdm optical communication networks: progress and challenges. *IEEE Journal on Selected Areas in communications*, 18(10):1810–1824, 2000.

[181] Nicholas Nethercote and Julian Seward. Valgrind: a framework for heavyweight dynamic binary instrumentation. *ACM Sigplan notices*, 42(6):89–100, 2007.

[182] Rolf Neugebauer, Gianni Antichi, José Fernando Zazo, Yury Audzevich, Sergio López-Buedo, and Andrew W Moore. Understanding PCIe performance for end host networking. In *SIGCOMM*, 2018.

[183] Kengo Nozaki, Akihiko Shinya, Shinji Matsuo, Yasumasa Suzaki, Toru Segawa, Tomonari Sato, Yoshihiro Kawaguchi, Ryo Takahashi, and Masaya Notomi. Ultralow-power all-optical RAM based on nanocavities. 2012.

[184] NVIDIA. Developting a linux kernel module using rdma for gpudirect. `https://docs.nvidia.com/cuda/pdf/GPUDirect_RDMA.pdf`. [Online; accessed 22-February-2020].

[185] NVIDIA. NVIDIA DGX-1. `https://www.nvidia.com/en-us/data-center/dgx-1/`. [Online; accessed 22-February-2020].

[186] NVIDIA. NVIDIA DGX-2. `https://www.nvidia.com/en-us/data-center/dgx-2/`. [Online; accessed 22-February-2020].

[187] NVIDIA. Nvidia gpudirect. `https://developer.nvidia.com/gpudirect`. [Online; accessed 22-February-2020].

[188] NVIDIA. Nvidia nvswitch. `http://images.nvidia.com/content/pdf/nvswitch-technical-overview.pdf`. [Online; accessed 22-February-2020].

[189] NVIDIA. NVIDIA Tesla V100 GPU Architecture. `https://images.nvidia.com/content/volta-architecture/pdf/volta-architecture-whitepaper.pdf`. [Online; accessed 22-February-2020].

[190] Masaaki Ono, Masanori Hata, Masato Tsunekawa, Kengo Nozaki, Hisashi Sumikura, Hisashi Chiba, and Masaya Notomi. Ultrafast and energy-efficient all-optical switching with graphene-loaded deep-subwavelength plasmonic waveguides. *Nature Photonics*, 14(1):37–43, 2020.

[191] Kishore Padmaraju, Dylan F Logan, Takashi Shiraishi, Jason J Ackert, Andrew P Knights, and Keren Bergman. Wavelength locking and thermally stabilizing microring resonators using dithering signals. *Journal of Lightwave Technology*, 2013.

[192] Gagandeep Panwar, Da Zhang, Yihan Pang, Mai Dahshan, Nathan DeBardeleben, Binoy Ravindran, and Xun Jian. Quantifying Memory Underutilization in HPC Systems and Using it to Improve Performance via Architecture Support. In *MICRO*, 2019.

[193] Antonios D Papaioannou, Reza Nejabati, and Dimitra Simeonidou. The benefits of a disaggregated data centre: A resource allocation approach. In *GLOBECOM*, 2016.

[194] J. T. Pawlowski. Hybrid memory cube (hmc). In *HOTCHIPS*, 2011.

[195] Erez Perelman, Greg Hamerly, and Brad Calder. Picking statistically valid and early simulation points. *PACT-03*.

[196] T Pinguet, B Analui, E Balmater, D Guckenberger, M Harrison, R Koumans, D Kucharski, Y Liang, G Masini, A Mekis, et al. Monolithically integrated high-speed cmos photonic transceivers. In *2008 5th IEEE international conference on group IV photonics*, pages 362–364. IEEE, 2008.

[197] Stelios Pitris, Christos Vagionas, George T Kanellos, Nikos Pleros, Rifat Kisacik, Tolga Tekin, and Ronald Broeke. Monolithically integrated all-optical SOA-based SR Flip-Flop on InP platform. In *Photonics in Switching (PS), 2015 International Conference on*, 2015.

[198] Stelios Pitris, Christos Vagionas, Tolga Tekin, Ronald Broeke, George Kanellos, and Nikos Pleros. WDM-enabled Optical RAM at 5 Gb/s Using a Monolithic InP Flip-Flop Chip. *Photonics Journal, IEEE*, 2016.

[199] N Pleros, D Apostolopoulos, D Petrantonakis, C Stamatiadis, and H Avramopoulos. Optical Static RAM Cell. *IEEE Photonics Technology Letters*, 2008.

[200] C Pollock, F Pardo, M Imboden, and DJ Bishop. Open loop control theory algorithms for high-speed 3d mems optical switches. *Optics express*, 28(2):2010–2019, 2020.

[201] R. Polster, Y. Thonnart, G. Waltener, J. Gonzalez, and E. Cassan. Efficiency optimization of silicon photonic links in 65-nm CMOS and 28-nm FDSOI technology nodes. *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, 2016.

[202] Filippo Ponzini, Fabio Cavaliere, Gianluca Berrettini, Marco Presi, Ernesto Ciaramella, Nicola Calabretta, and Antonella Bogoni. Evolution scenario toward wdm-pon. *Journal of Optical Communications and Networking*, 1(4):C25–C34, 2009.

[203] Claudio Porzi, Fabio Falconi, Giorgia Parca, Luigi Ansalone, Paolo Ghelfi, and Antonella Bogoni. Fast-reconfigurable microwave photonics phase shifter using silicon microring resonators. *IEEE Journal of Quantum Electronics*, 57(1):1–9, 2021.

[204] Roberto Proietti, Yawei Yin, Zheng Cao, CJ Nitta, V Akella, and SJ Ben Yoo. Low-latency interconnect optical network switch (LIONS). In *Optical Switching in Next Generation Data Centers*. 2018.

[205] Mateja Putic, Swagath Venkataramani, Schuyler Eldridge, Alper Buyuktosunoglu, Pradip Bose, and Mircea Stan. Dyhard-dnn: Even more dnn acceleration with dynamic hardware reconfiguration. In *Proceedings of the 55th Annual Design Automation Conference*, DAC '18, New York, NY, USA, 2018. Association for Computing Machinery.

[206] Lei Qiao, Weijie Tang, and Tao Chu. 32× 32 silicon electro-optic switch with built-in monitors and balanced-status units. *Scientific Reports*, 7(1):1–7, 2017.

[207] Luiz E. Ramos, Eugene Gorbatov, and Ricardo Bianchini. Page Placement in Hybrid Memory Systems. In *ICS*, 2011.

[208] Preeti Rani, Yogita Kalra, and RK Sinha. Design and analysis of polarization independent all-optical logic gates in silicon-on-insulator photonic crystal. *Optics Communications*, 374:148–155, 2016.

[209] Graham T Reed, G Mashanovich, F Yand Gardes, and DJ Thomson. Silicon optical modulators. *Nature photonics*, 4(8):518–526, 2010.

[210] Charles Reiss, Alexey Tumanov, Gregory R Ganger, Randy H Katz, and Michael A Kozuch. Towards Understanding Heterogeneous Clouds at Scale: Google Trace Analysis. *ISTCCC, Tech. Rep*, 2012.

[211] Albert Reuther, Peter Michaleas, Michael Jones, Vijay Gadepally, Siddharth Samsi, and Jeremy Kepner. Survey and benchmarking of machine learning accelerators. *arXiv preprint arXiv:1908.11348*, 2019.

[212] C Rios, M Stegmaier, P Hosseini, D Wang, and T Scherer. Integrated all-photonic non-volatile multi-level memory. *Nature ...*, 2015.

[213] Carlos Ríos, Nathan Youngblood, Zengguang Cheng, Manuel Le Gallo, Wolfram HP Pernice, C David Wright, Abu Sebastian, and Harish Bhaskaran. All-photonic in-memory computing based on phase-change materials. In *CLEO: Science and Innovations*, pages SM2J–2. Optical Society of America, 2019.

[214] Thomas B Rolinger, Tyler A Simon, and Christopher D Krieger. An empirical evaluation of allgatherv on multi-gpu systems. In *2018 18th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing (CCGRID)*, pages 123–132. IEEE, 2018.

[215] F. Rosenblatt. Perceptron simulation experiments. *Proceedings of the IRE*, 48(3):301–309, March 1960.

[216] Sébastien Rumley, Meisam Bahadori, Ke Wen, Dessislava Nikolova, and Keren Bergman. Phoenixsim: Crosslayer design and modeling of silicon photonic interconnects. In *AISTECS*, 2016.

[217] Sébastien Rumley, Dessislava Nikolova, Robert Hendry, Qi Li, David Calhoun, and Keren Bergman. Silicon photonics for exascale systems. *J. Lightwave Technol.*, 2015.

[218] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. Imagenet large scale visual recognition challenge, 2014.

[219] Viktoriia Rutckaia and Joerg Schilling. Ultrafast low-energy all-optical switching. *Nature Photonics*, 14(1):4–6, 2020.

[220] Aryabartta Sahu et al. Performance and area trade-off of 3d-stacked dram based chip multiprocessor with hybrid interconnect. *IEEE Transactions on Emerging Topics in Computing*, 2019.

[221] Bahaa EA Saleh and Malvin Carl Teich. *Fundamentals of photonics*. john Wiley & sons, 2019.

[222] Daniel Sanchez and Christos Kozyrakis. ZSim: Fast and accurate microarchitectural simulation of thousand-core systems. In *ISCA*, 2013.

[223] Ken-ichi Sato. Realization and application of large-scale fast optical circuit switch for data center networking. *Journal of Lightwave Technology*, 2018.

[224] Carsten Schinke, P Christian Peest, Jan Schmidt, Rolf Brendel, Karsten Bothe, Malte R Vogt, Ingo Kröger, Stefan Winter, Alfred Schirmacher, Siew Lim, et al. Uncertainty analysis for the coefficient of band-to-band absorption of crystalline silicon. *AIP Advances*, 5(6):067168, 2015.

[225] A. Sebastian, I. Boybat, M. Dazzi, I. Giannopoulos, V. Jonnalagadda, V. Joshi, G. Karunaratne, B. Kersting, R. Khaddam-Aljameh, S. R. Nandakumar, A. Petropoulos, C. Piveteau, T. Antonakopoulos, B. Rajendran, M. L. Gallo, and E. Eleftheriou. Computational memory-based inference and training of deep neural networks. In *2019 Symposium on VLSI Technology*, pages T168–T169, June 2019.

[226] Assaf Shacham, Keren Bergman, and Luca P Carloni. Photonic networks-on-chip for future generations of chip multiprocessors. *IEEE Transactions on Computers*, 57(9):1246–1260, 2008.

[227] J. Shalf. Hpc interconnects at the end of moore's law. In *2019 Optical Fiber Communications Conference and Exhibition (OFC)*, pages 1–3, March 2019.

[228] Sayeh Sharify, Alberto Delmas Lascorz, Mostafa Mahmoud, Milos Nikolic, Kevin Siu, Dylan Malone Stuart, Zissis Poulos, and Andreas Moshovos. Laconic deep learning inference acceleration. In *Proceedings of the 46th International Symposium on Computer Architecture*, pages 304–317, 2019.

[229] Y. Shen, X. Meng, Q. Cheng, S. Rumley, N. Abrams, A. Gazman, E. Manzhosov, M. S. Glick, and K. Bergman. Silicon Photonics for Extreme Scale Systems. *JLT*, 2019.

[230] Y. Shen, X. Meng, Q. Cheng, S. Rumley, N. Abrams, A. Gazman, E. Manzhosov, M. S. Glick, and K. Bergman. Silicon photonics for extreme scale systems. *Journal of Lightwave Technology*, 2019.

[231] Yichen Shen, Nicholas C Harris, Scott Skirlo, Mihika Prabhu, Tom Baehr-Jones, Michael Hochberg, Xin Sun, Shijie Zhao, Hugo Larochelle, Dirk Englund, et al. Deep learning with coherent nanophotonic circuits. *Nature Photonics*, 11(7):441, 2017.

[232] Yiwen Shen, Xiang Meng, Qixiang Cheng, Sébastien Rumley, Nathan Abrams, Alexander Gazman, Evgeny Manzhosov, Madeleine Strom Glick, and Keren Bergman. Silicon photonics for extreme scale systems. *Journal of Lightwave Technology*, 37(2):245–259, 2019.

[233] David Silver, Aja Huang, Chris J. Maddison, Arthur Guez, Laurent Sifre, George van den Driessche, Julian Schrittwieser, Ioannis Antonoglou, Veda Panneershelvam, Marc Lanctot, Sander Dieleman, Dominik Grewe, John Nham, Nal Kalchbrenner, Ilya Sutskever, Timothy Lillicrap, Madeleine Leach, Koray Kavukcuoglu, Thore Graepel, and Demis Hassabis. Mastering the game of go with deep neural networks and tree search. *Nature*, 529(7587):484–489, 2016.

[234] TOP500 Supercomputer Sites. Top500 november 2019 list, 2019.

[235] Meint Smit, Xaveer Leijtens, Huub Ambrosius, Erwin Bente, Jos van der Tol, Barry Smalbrugge, Tjibbe de Vries, Erik-Jan Geluk, Jeroen Bolk, Rene van Veldhoven, Luc Augustin, Peter Thijs, Domenico D'Agostino, Hadi Rabbani, Katarzyna Lawniczuk, Stanislaw Stopinski, Saeed Tahvili, Antonio Corradi, Emil Kleijn, Dzmitry Dzibrou, Manuela Felicetti, Elton Bitincka, Valentina Moskalenko, Jing Zhao, Rui Santos, Giovanni Gilardi, Weiming Yao, Kevin Williams, Patty Stabile, Piet Kuindersma, Josselin Pello, Srivathsa Bhat, Yuqing Jiao, Dominik Heiss, Gunther Roelkens, Mike Wale, Paul Firth, Francisco Soares, Norbert Grote, Martin Schell, Helene Debregeas, Mohand Achouche, Jean-Louis Gentner, Arjen Bakker, Twan Korthorst, Dominic Gallagher, Andrew Dabbs, Andrea Melloni, Francesco Morichetti, Daniele Melati, Adrian Wonfor, Richard Penty, Ronald Broeke, Bob Musk, and Dave Robbins. An introduction to inp-based generic integration technology. *Semiconductor Science and Technology*, 2014.

[236] Meint Smit, JJGM Van der Tol, and Martin Hill. Moore's law in photonics. *Laser & Photonics Reviews*, 6(1):1–13, 2012.

[237] Meint Smit, Kevin Williams, and Jos van der Tol. 1.3 integration of photonics and electronics. In *2019 IEEE International Solid-State Circuits Conference-(ISSCC)*, pages 29–34. IEEE, 2019.

[238] Shihao Song, Anup Das, Onur Mutlu, and Nagarajan Kandasamy. Enabling and exploiting partition-level parallelism (palp) in phase change memories. *ACM Transactions on Embedded Computing Systems (TECS)*, 18(5s):1–25, 2019.

[239] M Stegmaier, C Rios, P Hosseini, C David Wright, H Bhaskaran, and W H P Pernice. All-photonic nonvolatile memory cells using phase-change materials. *2015 IEEE Photonics Conference (IPC)*, 2015.

[240] Brian Stern, Xingchen Ji, Yoshitomo Okawachi, Alexander L Gaeta, and Michal Lipson. Battery-operated integrated frequency comb generator. *Nature*, 562(7727):401–405, 2018.

[241] Chen Sun, Mark T Wade, Yunsup Lee, Jason S Orcutt, Luca Alloatti, Michael S Georgas, Andrew S Waterman, Jeffrey M Shainline, Rimas R Avizienis, Sen Lin, Benjamin R Moss, Rajesh Kumar, Fabio Pavanello, Amir H Atabaki, Henry M Cook, Albert J Ou, Jonathan C Leu, Yu-Hsin Chen, Krste Asanovic, Rajeev J Ram, Miloš A Popović, and Vladimir M Stojanović. Single-chip microprocessor that communicates directly using light. *Nature ...*, 2015.

[242] Vivienne Sze, Yu-Hsin Chen, Joel Emer, Amr Suleiman, and Zhengdong Zhang. Hardware for machine learning: Challenges and opportunities. In *2017 IEEE Custom Integrated Circuits Conference (CICC)*, pages 1–8. IEEE, 2017.

[243] Yaniv Taigman, Ming Yang, Marc'Aurelio Ranzato, and Lior Wolf. Deepface: Closing the gap to human-level performance in face verification. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1701–1708, 2014.

[244] Sasha Targ, Diogo Almeida, and Kevin Lyman. Resnet in resnet: Generalizing residual architectures. *arXiv preprint arXiv:1603.08029*, 2016.

[245] M. Y. Teh, Z. Wu, and K. Bergman. Flexspander: augmenting expander networks in high-performance systems with optical bandwidth steering. *IEEE/OSA Journal of Optical Communications and Networking*, 12(4):B44–B54, April 2020.

[246] Min Teng, Amirmahdi Honardoost, Yousef Alahmadi, Sajad Saghaye Polkoo, Keisuke Kojima, He Wen, C Kyle Renshaw, Patrick LiKamWa, Guifang Li, Sasan Fathpour, et al. Miniaturized silicon photonics devices for integrated optical signal processors. *Journal of Lightwave Technology*, 38(1):6–17, 2020.

[247] Thomas N. Theis and H.-S. Philip Wong. The end of moore's law: A new beginning for information technology. *Computing in Science & Engineering*, 2017.

[248] Martin Thomaschewski, Vladimir A Zenin, Christian Wolff, and Sergey I Bozhevolnyi. Plasmonic monolithic lithium niobate directional coupler switches. *Nature communications*, 11(1):1–6, 2020.

[249] David Thomson, Aaron Zilkie, John E Bowers, Tin Komljenovic, Graham T Reed, Laurent Vivien, Delphine Marris-Morini, Eric Cassan, Léopold Virot, Jean-Marc Fédéli, et al. Roadmap on silicon photonics. *Journal of Optics*, 2016.

[250] Christos A Thraskias, Eythimios N Lallas, Niels Neumann, Laurent Schares, Bert J Offrein, Ronny Henker, Dirk Plettemeier, Frank Ellinger, Juerg Leuthold, and Ioannis Tomkos. Survey of photonic and plasmonic interconnect technologies for intra-datacenter and high-performance computing communications. *IEEE Commun. Surv. Tutor.*, 2018.

[251] Rahat Ullah, Bo Liu, Qi Zhang, Muhammad Saad Khan, Ibrar Ahmad, Amjad Ali, Razaullah Khan, Qinghua Tian, Cheng Yan, and Xiangjun Xin. Pulsed laser-based optical frequency comb generator for high capacity wavelength division multiplexed passive optical network supporting 1.2 tbps. *Optical Engineering*, 55(9):096106, 2016.

[252] Christos Vagionas, Dimitrios Fitsios, George T Kanellos, Nikos Pleros, and Amalia Miliou. Optical RAM and Flip-Flops Using Bit-Input Wavelength Diversity and SOA-XGM Switches. *Lightwave Technology, Journal of*, 2012.

[253] Joris Van Campenhout, William M Green, Solomon Assefa, and Yurii A Vlasov. Low-power, 2x2 silicon electro-optic switch with 110-nm bandwidth for broadband reconfigurable optical networks. *Opt. Express, OE*, 2009.

[254] Yurii A Vlasov. Silicon cmos-integrated nano-photonics for computer and data communications beyond 100g. *IEEE Communications Magazine*, 50(2):s67–s72, 2012.

[255] Zhechao Wang, Amin Abbasi, Utsav Dave, Andreas De Groote, Sulakshna Kumari, Bernadette Kunert, Clement Merckling, Marianna Pantouvaki, Yuting Shi, Bin Tian, et al. Novel light source integration approaches for silicon photonics. *Laser & Photonics Reviews*, 11(4):1700063, 2017.

[256] Jonas Weiss, Roger Dangel, Jens Hofrichter, Folkert Horst, Daniel Jubin, Norbert Meier, Antonio La Porta, and Bert Jan Offrein. Optical Interconnects for Disaggregated Resources in Future Datacenters. In *ECOC*, 2014.

[257] Ke Wen, Payman Samadi, Sébastien Rumley, Christine P Chen, Yiwen Shen, Meisam Bahadori, Keren Bergman, and Jeremiah Wilke. Flexfly: Enabling a reconfigurable dragonfly through silicon photonics. In *SC'16: Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis*, pages 166–177. IEEE, 2016.

[258] Samuel Williams, Andrew Waterman, and David Patterson. Roofline: an insightful visual performance model for multicore architectures. *Communications of the ACM*, 52(4):65–76, 2009.

[259] Alan Willner. *Optical fiber telecommunications*, volume 11. Academic Press, 2019.

[260] Gregory L Wojcik, Dongliang Yin, Alexey R Kovsh, Alexey E Gubenko, Igor L Krestnikov, Sergey S Mikhrin, Daniil A Livshits, David A Fattal, Marco Fiorentino, and Raymond G Beausoleil. A single comb laser source for short reach wdm interconnects. In *Novel in-Plane Semiconductor Lasers Viii*, volume 7230, page 72300M. International Society for Optics and Photonics, 2009.

[261] Carole-Jean Wu, David Brooks, Kevin Chen, Douglas Chen, Sy Choudhury, Marat Dukhan, Kim Hazelwood, Eldad Isaac, Yangqing Jia, Bill Jia, et al. Machine learning at facebook: Understanding inference at the edge. In *2019 IEEE International Symposium on High Performance Computer Architecture (HPCA)*, pages 331–344. IEEE, 2019.

[262] Y. Xie, Y. Shi, L. Liu, J. Wang, R. Priti, G. Zhang, O. Liboiron-Ladouceur, and D. Dai. Thermally-reconfigurable silicon photonic devices and circuits. *IEEE Journal of Selected Topics in Quantum Electronics*, 26(5):1–20, 2020.

[263] Qianfan Xu, Bradley Schmidt, Sameer Pradhan, and Michal Lipson. Micrometre-scale silicon electro-optic modulator. *nature*, 435(7040):325–327, 2005.

[264] Yan Yan, George M Saridis, Yi Shu, Bijan Rahimzadeh Rofoee, Shuangyi Yan, Murat Arslan, Thomas Bradley, Natalie V Wheeler, Nicholas Heng-Loong Wong, Francesco Poletti, et al. All-optical Programmable Disaggregated Data Centre Network Realized by FPGA-based Switch and Interface Card. *JLT*, 2016.

[265] Chenran Ye, Ke Liu, Richard A Soref, and Volker J Sorger. A compact plasmonic MOS-based 2x2 electro-optic switch. *arXiv.org*, 2015.

[266] H. Yoon, J. Meza, R. Ausavarungnirun, R. A. Harding, and O. Mutlu. Row Buffer Locality Aware Caching Policies for Hybrid Memories. In *ICCD*, 2012.

[267] Hanbin Yoon, Justin Meza, Naveen Muralimanohar, Norman P Jouppi, and Onur Mutlu. Efficient Data Mapping and Buffering Techniques for Multilevel Cell Phase-Change Memories. *TACO*, 2014.

[268] Yosia, Akihiko Shinya, Eiichi Kuramochi, Masaya Notomi, Shinji Matsuo, Takaaki Kakitsuka, Takasumi Tanabe, and Tomonari Sato. All-optical on-chip bit memory based on ultra high Q InGaAsP photonic crystal. *Optics Express*, 2008.

[269] Xiangyao Yu, Christopher J Hughes, Nadathur Satish, Onur Mutlu, and Srinivas Devadas. Banshee: Bandwidth-efficient DRAM Caching Via Software/hardware Cooperation. In *MICRO*, 2017.

[270] Anton V Zasedatelev, Anton V Baranikov, Darius Urbonas, Fabio Scafirimuto, Ullrich Scherf, Thilo Stöferle, Rainer F Mahrt, and Pavlos G Lagoudakis. A room-temperature organic polariton transistor. *Nature Photonics*, 13(6):378–383, 2019.

[271] Georgios Zervas, Hui Yuan, Arsalan Saljoghei, Qianqiao Chen, and Vaibhawa Mishra. Optically disaggregated data centers with minimal remote memory latency: Technologies, architectures, and resource allocation. *J. Opt. Commun. Netw.*, 2018.

[272] Chong Zhang, Shangjian Zhang, Jon D Peters, and John E Bowers. 8× 8× 40 gbps fully integrated silicon photonic network on chip. *Optica*, 3(7):785–786, 2016.

[273] Y. Zhang, X. Xiao, K. Zhang, S. Li, A. Samanta, Y. Zhang, K. Shang, R. Proietti, K. Okamoto, and S. J. B. Yoo. Foundry-enabled scalable all-to-all optical interconnects using silicon nitride arrayed waveguide router interposers and silicon photonic transceivers. *IEEE Journal of Selected Topics in Quantum Electronics*, 25(5):1–9, 2019.

[274] Z. Zhu, G. Di Guglielmo, Q. Cheng, M. Glick, J. Kwon, H. Guan, L. P. Carloni, and K. Bergman. Photonic switched optically connected memory: An approach to address memory challenges in deep learning. *Journal of Lightwave Technology*, 38(10):2815–2825, 2020.

[275] Ziyi Zhu, Yiwen Shen, Yishen Huang, Alexander Gazman, Maarten Hattink, and Keren Bergman. Flexible Resource Allocation Using Photonic Switched Interconnects for Disaggregated System Architectures. In *OFC*, 2019.

[276] Amir Kavyan Ziabari, Yifan Sun, Yenai Ma, Dana Schaa, José L Abellán, Rafael Ubal, John Kim, Ajay Joshi, and David Kaeli. Umh: A hardware-based unified memory hierarchy for systems with multiple discrete gpus. *ACM Transactions on Architecture and Code Optimization (TACO)*, 13(4):1–25, 2016.

[277] Dimitrios Ziakas, Allen Baum, Robert A Maddox, and Robert J Safranek. Intel® quickpath interconnect architectural features supporting scalable system architectures. In *2010 18th IEEE Symposium on High Performance Interconnects*, pages 1–6. IEEE, 2010.