Predict, Share, and Recycle your Way to Low Power Nanophotonic Networks

JANIBUL BASHIR, Department of Computer Science and Engineering, Indian Institute of Technology, New Delhi, India
SMRUTI RANJAN SARANGI, Department of Computer Science and Engineering, Indian Institute of Technology, New Delhi, India

High static power consumption is widely regarded as one of the largest bottlenecks in creating scalable optical NoCs. The standard techniques to reduce static power are based on sharing optical channels, and modulating the laser. We show in this paper that state of the art techniques in these areas are suboptimal, and there is a significant room for further improvement. We propose two novel techniques – a neural network based method for laser modulation by predicting optical traffic, and a distributed and altruistic algorithm for channel sharing – that are significantly closer to a theoretically ideal scheme. In spite of this, a lot of laser power still gets wasted. We propose to reuse this energy to heat micro-ring resonators (achieve thermal tuning) by efficiently recirculating it. These three methods help us significantly reduce the energy requirements. Our design consumes 4.7X lower laser power as compared to other state of the art proposals. In addition, it results in a 31% improvement in performance and 39% reduction in ED^2 for a suite of Splash2 and Parsec benchmarks.

CCS Concepts: • **Networks → Photonic Network on chip**; • **Computer systems organization → Optical Interconnection architectures**; **System on a chip**.

Additional Key Words and Phrases: Nanophotonics, on-chip networks, static power consumption, neural networks, ring resonators

ACM Reference Format:
Janibul Bashir and Smruti Ranjan Sarangi. 2019. Predict, Share, and Recycle your Way to Low Power Nanophotonic Networks. *ACM J. Emerg. Technol. Comput. Syst.* 1, 1, Article 1 (January 2019), 25 pages. https://doi.org/10.1145/3356585

1 INTRODUCTION

The last ten years have seen a lot of research in optical nanophotonic networks [5, 6, 76, 80]. As of today, most of the basic devices have been fabricated, and their efficacy and superiority over conventional network elements has been demonstrated [45, 58]. However, to harness the natural advantages of optical networks – low latency, and high bandwidth – it will take the industry some more time. The latest road maps from Intel [64], and IBM [63] project that on-chip optical networks will become feasible by the end of this decade. Test chips with 850 optical components have already been fabricated by Sun et al. [70]. Recently, a 2-core chip on an $18mm^2$ die with a photonics based network was successfully demonstrated in UC Berkeley [48]. The on-chip bandwidth is a commendable 300 Gbps per square millimeter. Furthermore, commercial fabs such as Circuits Multi

Authors' addresses: Janibul Bashir, Department of Computer Science and Engineering, Indian Institute of Technology, New Delhi, India, janibbashir@cse.iitd.ac.in; Smruti Ranjan Sarangi, Department of Computer Science and Engineering, Indian Institute of Technology, New Delhi, India, srsarangi@cse.iitd.ac.in.

1550-4832/2019/1-ART1 $15.00
https://doi.org/10.1145/3356585

Projects (CMP) [60] have started offering fabrication services for multi-layer silicon photonic ICs. They can fabricate most common passive and active optical structures using a 300 nm based SOI process.

HP has recently released the prototype of a supercomputer called 'The Machine' that uses interconnects based on photonics. It uses optical fibers to communicate between the nodes inside the system [3]. Similarly, Intel has been working for the last 10 years to develop on-chip photonic components [69], and in pursuance of this goal it has already released a board level photonics solution (Optical PCIX [32]); the next step is to integrate such solutions on-chip.

The advantages of optical networks arise from the fundamental nature of photons. They can travel faster (5X faster than signals in electrical networks), and it is possible to multiplex photons at different wavelengths on the same optical channel (WDM). The latter allows for significantly more bandwidth. The other advantages include data rate and energy consumption almost independent of on-chip distance, low power dissipation, and reduced interference. Insofar as such advantages are concerned, there is no divergence of opinions about the advantages that will accrue from a practical realization of on-chip photonics. Numerous authors [53, 73] have compared optical networks with electrical networks, and have reported on an average 30-50% improvement in performance.

However, optical networks are not a panacea to all our problems. They still have significant issues with respect to high static power consumption. This again arises because of the basic nature of photons. Photons cannot be stored easily, and thus need to be transmitted continuously. We end up losing a lot of power in this process. There are proposals in literature that have used lasers on the chip and it is possible to switch such lasers on-demand [16, 42]. However, on-chip lasers have several disadvantages such as low wall plug efficiency, complete heat dissipation inside the chip and many more [6]. Moreover, when the same fast-switched lasers are used off-chip, there is an additional latency of 2-4 cycles because we need to compute the power requirement in the immediate future, and communicate this to the off-chip laser.

The standard approach used in prior work [5–7] while using off-chip lasers is to *predict* and *share*. We predict periods of inactivity within the chip and turn off the lasers, and then optimally share the light between the optical stations (transmitter + receiver) during periods of activity. Predicting network activity in the future can be done by looking at network activity in the past [7, 82], PCs of memory instructions [57], and directory misses [47]. In addition, a novel machine learning based approach also proves helpful in predicting such activity [76]. Similarly, sharing the generated optical power efficiently among stations helps us use unused optical power, and also helps us reduce the number of waveguides (optical channels). By reducing the number of waveguides, we proportionately reduce the number of associated optical filters (ring resonators), and the micro-heaters required to maintain them at a fixed temperature. The power to run these micro-heaters (tuning power [49]) is a major component of the overall static power dissipation.

Reducing static power dissipation in optical networks is a very established area with a rich body of literature. Our approach in this paper, is to take some of the best performing works in this area, and create a scheme – *PShaRe*– that outperforms them by at least 20%. We observed that to create such a scheme we need to find novel ways for the *predict* and *share* part, and also introduce a new paradigm: *recycle* (reuse the wasted energy thrown out of the chip). Our approach is thus fundamentally different – predict-share-recycle.

Let us briefly enumerate our key contributions here.

(1) **ANN based Predictor :** We introduce a novel mechanism for predicting the network traffic with the help of an artificial neural network (ANN). We claim that non-linear prediction with the help of ANNs is better suited for predicting the network traffic in on-chip networks as it

produces far more accurate predictions, and is singularly responsible for 2.5X reduction in laser power (as compared to state of the art proposals).

(2) **Distributed Arbitration Scheme :** For *sharing* power, we deviate from the conventional approach [52] of using token based arbitration, and use a novel, altruistic, and power efficient distributed arbiter instead.

(3) **Power Reuse Scheme :** We introduce a new set of waveguides in the chip to recirculate the wasted laser energy such that it can be used to heat the microring resonators. We use plasmonics based devices to convert optical energy into heat. This approach helps us reduce tuning power by nearly 38%.

(4) **Evaluate :** With our new predict-share-recycle design, we evaluated a 32-core chip. For a suite of Splash2 and Parsec benchmarks, we were able to improve the performance by 31% simultaneously reduce ED^2 by 39% (as compared to the nearest state of the art competitor).

The natural question that arises in any work that uses emerging technologies is **feasibility**. In prior work, this concern has been addressed by using components that are either already commercially available, or whose prototype implementations have reached maturity [7, 36, 59, 65]. We use both these strategies in our paper, and only use components that are widely regarded as extremely promising, are preferably already commercially available, and also have a history of being heavily used in the architecture community. We have simulated all our optical components using industry standard simulators (RSoft [71] and Lumerical [43, 44]), and we have synthesized all the proposed electronic components in VHDL. Their area, power, and timing results are in the paper. Our work can be looked at as an addendum to an existing optical NoC, and is thus generic in nature.

Table 1 gives the treatise of the rest of the paper and summarize our contributions in each section.

Section #	Summary	Contributions	
2	Nanophotonics and ANN background		
3	Characterization and motivation	Derived Design Insights	
4	Design and Implementation	Section 4.2	ANN based Predictor for predicting on-chip traffic
		Section 4.3	A novel, altruistic and distributed arbitration scheme.
		Section 4.4	Using unused optical power to reduce tuning power
5	Evaluation	Evaluate our design decisions for Splash and Parsec suites	
6	Conclusion		

Table 1. Treatise

2 BACKGROUND

This section provides a brief overview of the important optical components relevant to our architecture.

2.1 Light Source

A laser is the most commonly used light source in on-chip photonic networks. The laser can be either off-chip [57] or integrated inside the chip [18]. The light produced by an off-chip laser is coupled into the chip using special tapered couplers [31] in order to decrease the coupling loss.

PShaRe uses off-chip lasers as the main light source. Specifically, we use an array of 32 directly modulated lasers(DML) [22] as an off-chip laser source. DML lasers can be modulated at GHz frequencies, and are very robust. These lasers are also thermally very stable [21, 22], and as of now are also commercially available. We use 32 such lasers so that we can create a virtual 32 step power source (by individually turning on/off lasers). In addition, *PShaRe* also uses a few on-chip lasers for arbitration. We can use any WDM compatible on-chip laser with fast switching time. InP and Ge based on-chip lasers are the best choices, as they have been fabricated already [13, 19] and can

be easily integrated on the chip. However, in our design we have used Ge based on-chip lasers as these lasers can easily operate at room temperature and are DWDM compatible [13, 42].

2.2 Tunable Power Splitters

Power splitters are required to divert optical power from the power waveguides. Each power splitter is associated with a power loss, which is a function of its split ratio. Cascading a large number of such splitters will result in an exponential increase in the optical power loss. Hence, to decrease the power loss, the optimal split ratio of the power splitters needs to be calculated. One such linear time approach was proposed by Peter et al. [56]. We use their method. To realize this we need a tunable splitter, whose split ratio can be changed dynamically (based on the power sourced by different components on the power waveguide). One such splitter based on a simple ring resonator has been proposed by Rajib et al. [23]. We use the same splitter (wide range version) in our design. These ring resonator based splitters require less than 400ps to retune.

2.3 Transmitter

The laser light is routed to the optical stations using a dedicated waveguide (optical channel) called a *power waveguide*. This light signal is monochromatic (at 1550nm). Individual stations source optical power from the power waveguide, whenever they need to transmit a message. Each station uses a ring resonator based comb splitter [39] to split the monochromatic light into 64 equispaced wavelengths. Subsequently, we modulate each signal with a ring resonator to encode information. This is a standard design, and has been used in many proposals [57, 73]. For modulating the optical signals, we use a microring resonator based modulator [78], which has low area and can modulate data at a rate of up to 10 GHz.

2.4 Receiver

At the receiving side, ring resonators filter out the modulated wavelengths from the data waveguide and guide them to an array of photodetectors. The photodetector converts this modulated light into a set of electrical signals using trans-impedance amplifiers. The area of the photodetector is $20\mu m^2$ and its capacitance is $1fF$ [73].

2.5 Optical Crossbar Based Topologies

Optical crossbar based topologies are preferred choices because they do not require complicated routing and path setup mechanisms. These are broadly classified into the following three categories in an N-station optical network: SWMR, MWSR, and MWMR.

In SWMR (single writer multiple reader) a station writes to only one waveguide, which is connected to the rest of the $N-1$ stations. A subset of these stations can read the data by sourcing a part of the incoming optical signal using beam splitters. Every station thus has to monitor $N-1$ other waveguides (one per sending station) for incoming messages. Firefly [53] optimizes this process by sending a reservation flit before the actual data transmission (R-SWMR) such that the desired receiver can be turned on. Receivers are off by default to save power. In the MWSR topology (multiple writer single reader), a station has a dedicated waveguide on which it receives data. The rest of the $N-1$ stations can write on that waveguide. Simultaneous writes are not allowed, and thus an arbitration mechanism is necessary. The MWMR topology combines SWMR and MWSR. This means that multiple stations can read or write on the same waveguide. The authors of Flexishare [52] have slightly constrained this paradigm by limiting access to a waveguide to only a subset of stations. This reduces the need for arbitration, and the number of ring resonators.

2.6 Artificial Neural Networks

To predict laser power in our optical NoC, we shall use an artificial neural network (ANN) that consists of interconnected processing elements that try to mimic the human brain [25]. Each such processing element is called a neuron and these neurons are connected together by weighted links. These neurons work together to provide a solution for a specific problem. ANNs are most commonly used in situations where we need to detect trends in the input data. They are highly beneficial in detecting non-linear patterns in the data, which are otherwise too complex to detect.

The ANNs comprise three different types of layers: input layer, one or more hidden layers and the output layer. The number of hidden layers and the number of neurons in each layer are determined experimentally. An ANN works in two different modes: 1) prediction mode and 2) training mode. In prediction mode each neuron takes an input and generates an output, which is fed to the next layer of neurons, whereas in training mode an ANN is provided with sample input/output values and then we adjust the weights of the links dynamically in order to achieve the desired output. The most commonly used algorithm in the training phase for adjusting the weights is the back propagation algorithm [28], which propagates the error backwards mostly based on the gradient descent method.

2.7 Photonic Heaters

For the energy recycling part, we shall use photonic heaters, which typically use metamaterials [4, 27]. There are free electrons in such materials, which absorb the injected light. The absorbed light increases the vibrations in the free charge carriers, resulting in increased collisions. The photons injected are thus eventually converted into heat. The heat generated diffuses and increases the surrounding temperature [24]. These photonic heaters should be ideally characterized by a very high photon absorption rate. The most common approach for creating such heaters is by using plasmonic nanoparticles typically made out of gold [74], silver [27], or Graphene [41]. We use the solution proposed by Lie et al. [27], who use silver nanoparticles. This has a photon absorption rate of more than 90%, and has an area of $768\mu m^2$. Such nanomaterials are already being used for treating tumors by the medical research community, and are commercially available. A lot of research [50, 81] has been done on embedding nanoparticles on a silicon surface. This is basically a two step process: charge the surface of the stamp that has protrusions laced with nanoparticles, and create an oxide layer on the silicon. Some nanoparticles will stick to the oxide layer after the stamp is removed.

3 MOTIVATION

In order to understand the behavior of various multi core benchmarks, we ran workloads from the Splash2 [77] and Parsec [12] benchmark suites on a chip with 32 4-issue out-of-order cores. We used the same set of workloads as [38, 57, 82]. The chip consists of 16MB of last level cache (L2) in the form of 32 cache banks. The architectural parameters of the chip are given in Table 4. We assume a hypothetical point-to-point topology connecting the cores and cache banks. In addition, we compared the power consumed by two state of the art proposals in this domain: *ColdBus* and *Probe*.

Probe [82]: *Probe* uses a prediction mechanism based on link and buffer utilization to modulate the laser.

ColdBus [57]: *ColdBus* is a SWMR crossbar based on-chip photonic network for 32 cores and 32 cache banks. It uses a prediction mechanism based on the PC of memory instructions, along with the past history of hits and misses.

The following workload characteristics led to the design of *PShaRe*.

3.1 High Laser Power Consumption → Predict

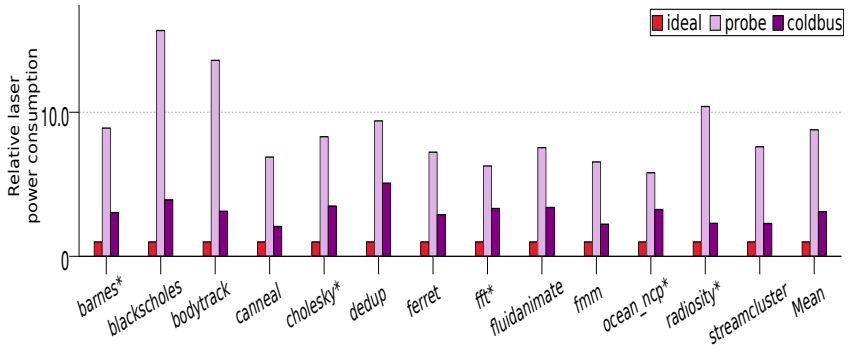

Fig. 1. Relative laser power consumption (* Splash2 workloads)

Figure 1 shows a comparison of the laser power consumption of two state of the art schemes (Probe [82] and ColdBus [57]), and an ideal scheme where we have one laser per optical transmitter – the laser is by default turned off. However, when the transmitter requires power, the laser is turned on instantaneously. We observe that Probe and ColdBus respectively consume 8.7X and 3.1X more laser power than the ideal scheme (See Figure 1). Thus, there is room for a more power efficient laser modulation scheme.

For better laser modulation, we need to accurately predict network activity. Prior work has assumed a linear dependence between network traffic in the future (number of messages) and two kinds of parameters: network parameters such as traffic in the last few epochs, and architectural parameters such as PC addresses, memory addresses, and queue occupancy. Our observations are to the contrary, and we saw a clear non-linear relationship between network/architectural parameters and network traffic. These observations are in line with those made by Khonsari et al. [35] and Bezzera et al. [10]. They have observed self-similar and highly super-linear patterns respectively. To capture such non-linear relationships we need to design efficient non-linear predictors (**Insight:Predict**).

3.2 Unbalanced Traffic → Share

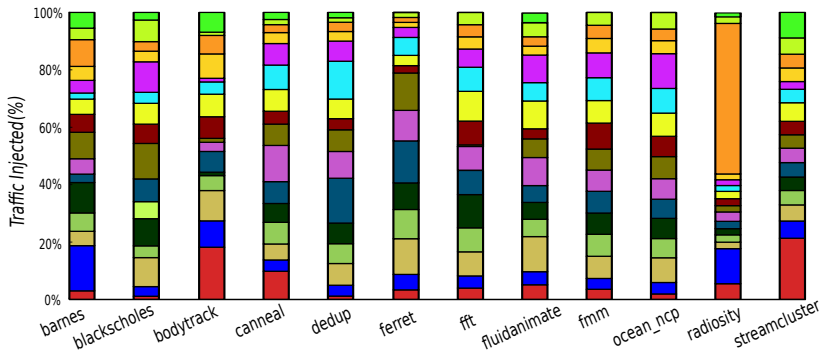

Fig. 2. Traffic injected across stations

Figure 2 shows the relative traffic injected by various optical stations in a network. Each color in the figure represents the percentage of traffic injected by an optical station. In almost every benchmark there are some stations which are highly active (at some point of time) and others

inject very little traffic in the network. In addition, we observed that the stations inject traffic in a burst mode – whenever a station starts injecting some traffic, it remains active for some time. As a result some stations require much more bandwidth and power at some points in their execution as compared to others. Thus, it will be highly beneficial if the data and power networks are shared. However, using a shared topology has its own disadvantages. One disadvantage is that there is a need for an arbitration scheme at the sender's side. Almost all the arbitration schemes proposed so far are based on optical tokens [52, 57]. However, such schemes are not power efficient because we need to continuously pump tokens into the network, resulting in high optical power consumption. The other disadvantage is that we have to broadcast a reservation flit before sending any message, in order to inform the interested stations to turn on their receivers [7, 52]. This too results in increased power consumption. Thus, there is a need for a power efficient arbitration scheme, which removes the need for continuously circulating tokens and broadcasting reservation flits (**Insight:Share**).

3.3 Energy Wasted → Recycle

Figure 1 shows that for two state of the art proposals, ColdBus [57] and Probe [82], nearly 67% and 88% of optical power supplied to the chip remains unused. Since it is practically impossible to design a predictor with 100% accuracy [57], there will always be some amount of unused power inside the chip. Let us thus use this energy for productive purposes notwithstanding losses in transporting it (waveguide losses) (**Insight:Recycle**).

4 DESIGN AND IMPLEMENTATION

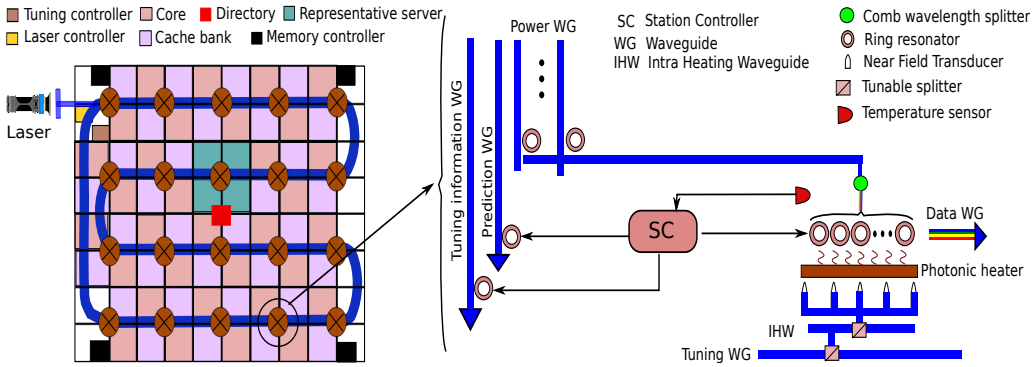

Fig. 3. Layout of the chip

4.1 Overview

In this paper, we consider a chip with 32 4-issue out-of-order cores and 32 cache banks. It uses a tiled architecture in which each tile consists of an optical station, two cores and two cache banks. Figure 3(a) shows the layout of the chip, comprising of waveguides and optical stations that form the optical network. The bundle of waveguides includes both the data and power waveguides, running parallel to each other. The light from the off-chip laser is divided into various parts using a cascading set of 16 tunable splitters [23], and routed to each station. There are a total of 16 power and 16 data channels and each channel is composed of 4 waveguides. Each data channel can carry 64 wavelengths using dense wavelength division multiplexing(DWDM) technology. Note that the ideas and insights behind the design of *PShaRe* are generic, and are not specific to this particular design.

Let us now proceed to describe the three novel mechanisms that are proposed in this paper: predict, share, and recycle.

The network traffic injected by the station in the last 5 epochs. This information is stored in a 5-entry shift register. Each such input is denoted as T_i (5 bits), where i is the index of the shift register. For example, T_1 denotes the network traffic in the most recent epoch.
The number of pending events(P) at the optical station (4 bits).
The number of private data cache evictions(E) in the current epoch (6 bits).

Table 2. Final set of parameters

4.2 Prediction Mechanism

We use an artificial neural network based predictor, which is one of the most commonly used non-linear predictors.

4.2.1 Design. The neural network used in our scheme is composed of three different layers: input layer, hidden layer and an output layer. The input features to the neural network include both the architectural and network related parameters. We start by using 14 different parameters as input to the ANN (please see online Appendix at [8]). Having higher number of features affect the overall prediction accuracy. However, at the cost of power, area and performance. Thus, there is a trade-off between the prediction accuracy and the area/power overheads. During our training and validation phase (please see [8]), we tried to eliminate the features that barely affect the overall prediction accuracy. Finally, we choose the most effective set of parameters as input to the predictor as given in Table 2.

We choose one hidden layer because every hidden layer increases the complexity and increases the prediction and training time. The number of neurons in the hidden layer is determined experimentally during the cross validation phase. We performed experiments with different numbers of neurons and found 6 to be the optimal value. Each neuron in the hidden layer implements the Sigmoid function in order to capture the non-linear trends in the network traffic. In the output layer, we have a single neuron implementing the threshold function. Thus, our neural network provides a Boolean output. Please refer to our online Appendix [8] for the feature selection, training, and cross validation of our predictor.

4.2.2 Prediction. *PShaRe* divides time into fixed size durations called *epochs* and predicts the laser power in advance for the next epoch. For each station, the prediction is binary. A value of '1' indicates that it is expected to transmit a message in the next epoch, and a value of '0' indicates complete inactivity during the next epoch. *False positives* are wasteful in terms of power, because the laser remains on when no messages are actually sent. Likewise, *false negatives* are hurtful in terms of performance because of the unavailability of laser power.

Each optical station is associated with a separate neural network based predictor as described in Section 4.2.1. Each neural network is trained at the beginning of an epoch only if there is a misprediction, and the prediction is performed at the end of every epoch. During the training phase, the optical station provides all the inputs that were used for predicting laser power in the previous epoch and the desired output. The ANN is trained iteratively till the weights are adjusted in such a way that the desired output is generated. For the prediction, the ANN is run at the end of every epoch. The optical station provides the required inputs and the ANN provides a Boolean output. If the output is '0', we predict that the station does not require power in the next epoch, whereas a value of '1' indicates that the station will generate network traffic and hence requires optical power.

4.2.3 Reconfiguring the Network. After the predictions are made by the optical stations, each station sends this information to the laser controller at the end of an epoch through a separate waveguide called the *prediction waveguide*. Each station uses a specific wavelength in the prediction waveguide to indicate its requirement (active low signalling). The predictions are sent to the laser controller, which collates the predictions and retunes the network at the beginning of the next epoch. The optical stations also sends this information to their respective representative servers, which we shall discuss next.

4.3 Arbitration Mechanism for Sharing Power

We have 20 stations (16 + additional stations for the directory and memory controllers), 16 power channels, and 16 data channels. Each power and data channel is assumed to be composed of 4 silicon waveguides. All the power and data channels are shared between the optical stations. An optical station can source power from any power channel. If it sources power from the i^{th} power channel, it can then use it to send a message on the i^{th} data channel: both are interlinked in this fashion. This process of sharing requires an arbitration process. For a group of 5 optical stations, we create a Representative Server (RS) physically located at the center of the chip, which brokers access to the shared waveguides on behalf of its constituent stations. Each RS is equivalent to a 2-port router with some extra memory. The RSs are connected to each other through an electrical ring based topology.

This cluster containing only the representative servers has a separate optical station, called the *Server Station*. The optical stations communicate with their respective representative servers through the reservation waveguides (all the waveguides run in parallel using a serpentine shaped layout). Each optical station and its respective representative server is allotted some specific wavelengths in the reservation waveguide in order to communicate with each other. This set of wavelengths allotted to this pair is called its reservation channel. Each such pair in the network is allotted a separate reservation channel, thus, resulting in a logical point-to-point links between the optical stations and their respective representative servers. Whenever an optical station is required to communicate with its representative server, it writes its data on the respective reservation channel. The *Server Station* reads the data from the channel and accordingly forwards the data to the respective representative server.

It should be noted that we can also use electrical point-to-point links. However, we require 20 such 7-bit wide electrical links, and this has the potential to create issues in routing. However, by using optical links, the same objective can be achieved using only three optical links (64 wavelengths in a single waveguide). Thus, keeping in view the uniformity, the power and area overheads, the ease of routing (with a serpentine structure), we choose optical links. Each station and each RS have a set of in-situ on-chip Ge based lasers [13, 42] (modulation time: 1 cycle). These lasers are used to generate optical power whenever a station and an RS want to communicate with each other. The RSs are connected with each other through an electrical network (because of close proximity and lower overheads). The additional overhead of having such on-chip lasers is small (0.3% in terms of area, and 3.3% in terms of total optical power).

4.3.1 Initialization. At the beginning of an epoch, the laser controller knows the number of channels, n_w, that should be carrying power (1. . .16 based on prediction in the previous epoch). It powers n_w channels using a network of cascaded tunable splitters [23]. Note that if the number of stations that wish to transmit in the current epoch is n_t, then $n_w = min(n_t, 16)$, because we have a maximum of 16 channels. Additionally, the RSs know their respective requirement (total requirement of their stations), and thus proceed with the allocation as follows. If $n_t > n_w$, then we allocate 4 channels to each RS. Otherwise, we allocate channels in this fashion: $1 \ldots r_1$ to RS1,

$r_1 + 1 \ldots (r_1 + r_2)$ to RS2 and so on. Here r_1 is the total requirement of the stations belonging to RS1, r_2 is the total requirement of RS2 and so on.

The channels allotted to an RS are called its local channels. Some times an RS might choose to lend a channel to another RS. For the other RS this channel will be a foreign channel. With each RS we have a 16-entry waveguide reservation table, WRT. Entry i of the table contains the following 5 bits: local/foreign (1 bit), free or busy (1 bit), and a timer (3 bits). When we allocate a channel, we set the timer. Once it reaches 0, the channel is deemed to be free.

4.3.2 Operation. Whenever a station wants to send data, it first sends a message to its RS. This message includes the receiver id (5 bits), and the number of flits in the message (2 bits). The RS accesses its WRT and tries to find if any of the available local waveguides are carrying optical power. If a local waveguide is available, then it is allotted to the optical station and the value of the timer is updated with the number of cycles for which the station needs the waveguide. In addition, the free/busy bit is set to 1. The timer is decremented every cycle, and when the value of the timer reaches 0, the waveguide becomes available again (free/busy bit = 0).

However, if no waveguide is available, the RS needs to take the help of other RSs. The algorithm for doing this is as follows. We maintain a lower triangular matrix with N row and N columns, where N stands for the number of RSs. Every time that RS i needs a waveguide it checks the status of the rest of RSs. If RS j has a free waveguide we increment $A[min(i, j), max(i, j)]$. After κ epochs we form pairs of RSs as follows. We first choose the largest value in the matrix, $A[i', j']$ and form a pair (i', j'). Then we set the $i'th$ and j'^{th} row and column to be 0. Then out of the remaining entries we find the largest value and so on. This ensures that we always pair RSs that are nearly 180° out of phase with each other (existence follows from **Insight:Share**). Every time that an RS needs a channel it asks its paired RS. If after λ cycles it does not get a channel, it sends a request to another randomly chosen pair (waits for λ cycles again). If there are an odd number of RSs, we will have a singleton. This RS needs to randomly ask other RSs for borrowing a channel if there is a need. Finally, note that the life of a set of pairs is limited to κ epochs, after which we create another set of pairs based on the values in the matrix A. Each RS stores the information regarding waiting optical stations (in its cluster) in a separate 5-entry table called the Pending Request Table (PRT). The RS allows other RSs (not a part of its pair) to use its allotted power channels only when their is no pending request in its PRT. However, for the RS that it is paired with, it gives it preferential status by serving at least one of its requests for every μ of its own requests. Please note that $\kappa = 2$, $\lambda = 4$, and $\mu = 4$ are the threshold values which have been selected based on the simulation results.

After allocating channel i to a station, the RS sends a reservation message to the receiver through its respective RS, indicating the data channel from which it needs to read data and the number of flits in the message. This eliminates the overhead of reservation messages [53]. The complete flow of operations for using the power channel is shown in Figure 4. There are no issues with scalability in this protocol. As we increase the number of RSs we create more and more pairs.

The simulation results show that the RSs send 13.1% of total power requests to the other RS in their respective pair, and our of these 63% of the requests are satisfied. This reduces the number of inter-RS messages to 37%.

4.4 Recycling Power

4.4.1 Tuning Network. In *PShaRe*, we divide the ring resonators into two groups: one group includes the ring resonators attached to the data waveguides (roughly 85%) and the remaining ring resonators belong to the other group (power waveguides and arbitration logic). In *PShaRe*, we tune the resonators in these groups separately. Recall that tuning is required to bring all the resonators to the same temperature because the resonant wavelength is a function of temperature.

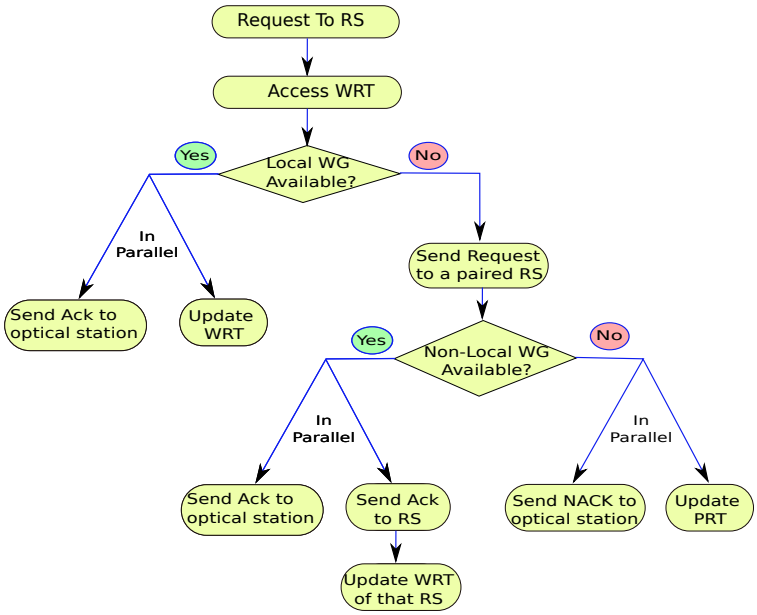

Fig. 4. Arbitration flow-chart

Our idea is to use the unused optical power to partially tune the ring resonators attached with the data waveguides. Note that all the ring resonators will still need to be connected to conventional microheaters [51]. The objective is to decrease the power supplied to the microheaters by reusing wasted optical power. Since heat is the lowest form of energy conversion of optical power to heat is easy and has a very high efficiency.

Resonators attached to the data waveguides are co-located, and thus they have a similar thermal profile [49]. This property allows us to place heaters at the center of such clusters (size:64 resonators), and target all of them at once. A separate set of waveguides called the tuning waveguides (TW) are used to carry the unused optical power. At the end of the power waveguides, the unused power is coupled into these waveguides. A cascaded set of Y-junctions [65] is used to combine the unused power into tuning waveguides (lengths adjusted appropriately for nullifying phase delay). As light travels through the TWs, the splitters attached split some portion of the light and send it towards each optical station. Within each station, the optical power is guided through a very small waveguide called the intra-heating waveguide (IHW) that distributes it to all the photonic heaters (see Section 2.7). We use near field transducers(NFT) [1] (See Figure 3(b)) to focus the light on the plasmonic material in the heater for maximum efficiency. The amount of optical power (P) required to heat a volume V of a material with volumetric heat capacity C by temperature T in time t is given by [11]:

$$P = \frac{T * C * V}{t} \tag{1}$$

4.4.2 Control System For Thermal Tuning. One important insight from [49] and our thermal simulations is that the thermal drift in co-located ring resonators is almost the same. It is well known that silicon is a thermal low pass filter [30], and thus we don't see appreciable temperature differences at the sub-micron scale. As a result, the ring resonators require the same amount of heat to bring them back to the ideal operating temperature. Keeping this in mind, we use a single

photonic heater for a group of 64 co-located ring resonators (Insight from [49]). Our photonic heater is a linear structure(see Section 2.7) placed beneath the group of ring resonators, whose job is to bring the temperature up to a given value, τ.

We use a diode based temperature sensor [17] for each cluster of 64 resonators with a proportional-integral-derivative (PID) based tuning controller (details in [8]). We implemented it using VHDL and synthesized it using the Cadence RTL compiler for 14 nm technology node. At the end of every epoch, optical stations send an average temperature deviation of its sensors from the nominal value (5-bit) to the tuning controller (TC) through a tuning information waveguide (TIW). We require 512 temperature sensors and photonic heaters to thermally tune the 32K ring resonators that we are considering. The area overhead is less than 0.5%.

The unused power in the TWs is distributed among the optical stations. Within each station, the incoming power is distributed equally among the photonic heaters. The station monitors the temperature of each group. At any point of time if the temperature of any constituent group rises above the threshold value, T_{max} (80 $^{\circ}C$), the Station Controller (SC) changes the split ratio of the splitter attached with the TWs in order to stop the incoming flow of optical power. It is a feedback control system in which the split ratio of the splitter is continuously changed based on the temperature of the ring resonators. The threshold value, T_{max}, is chosen to be a value that is less than the temperature τ (target ring resonator temperature, 85$^{\circ}C$). This helps us to make our control algorithm simple. The rest of the heating is done with the help of traditional micro-ring heaters [51].

There are two different methods of distributing the unused power among the optical stations. One method is to distribute the unused power equally among all the stations and then convert it into heat. This is very inefficient owing to large losses in splitters, crossings, and the waveguide itself. Our proposal is to dump the entire unused power equally at the stations closer to the start of the TWs. The number of such stations is equal to $5 \times \#total\ optical\ stations/Avg.\ deviation$. The average deviation is calculated at the end of every epoch and the reconfiguration is done at the beginning of an epoch.

Note that a region of silicon has a thermal resistance and capacitance (conceptually similar to their electrical analogs). Pumping heat is similar to pumping charge into a capacitor. The heat translates to increased temperature and the elevated temperature is maintained for a sizable duration of time (order of the thermal time constant, several ms). During this time we can reduce the electrical tuning power commensurately. Thus infrequently pumping in bursts of heat can effectively reduce tuning power for large durations of time because of the inherent thermal capacitance.

4.5 Laser Controller

The algorithm used by the laser controller is adapted from [57] with suitable modifications. The laser controller computes the following using a host of lookup tables: split ratio (4-bits) of each splitter in the network, and laser power (5 bits). It sends the output laser power (number of DML lasers to be turned on) to the laser array using a fast optical link. Table 3 (also see [57]) summarizes this discussion and shows the various actions that are required for reconfiguring the network and the time that they take. In an epoch, 12 cycles are required for the prediction and reconfiguration stage and out of these, the stations are inactive for only 1-3 cycles.

5 EVALUATION

5.1 Experimental Setup

For all our simulations we use the cycle accurate architectural simulator, Tejas [66], which has been rigorously validated with native hardware. It has extensive support for simulating optical

Actions			Cycles
Prediction Phase			
Neural Network Prediction			3
Send prediction to the laser controller	1	RSs waveguide allocation	3
Collate recommendations and predictions	2		
Calculate power requirements			2
Send msg. to the off-chip laser (E/O+tran.+O/E)			2
Reconfiguration Phase (Inactive Phase)			
Compute laser array config.	1	Table lookup	1
Laser retuning	1	Reconfigure splitters	1
Total			12

Table 3. Prediction and reconfiguration

Parameter	Value	Parameter	Value
Cores	32	Technology	14nm
Frequency	2.5 GHz		
Processor Core			
pipeline	Four-issue out-of-order	IW size	54
iTLB	128 entry	dTLB	128 entry
Private L1 i-cache, d-cache			
Write-mode	Write-back	Block size	64 bytes
Associativity	4	Size	64 kB
Latency	2 cycles	MSHRs	32
Shared L2			
Write-mode	Write-back	Block size	64 bytes
Associativity	4	# banks	32
Latency (per bank)	8 cycles	Bank size	512 KB
Main Memory			
Latency	150 cycles	Mem. controllers	4
Queue Sizes			
Optical Station Queue	16		
Electrical NoC			
Topology	Flattened Butterfly	Routing Alg.	X-Y
Flit size	16 bytes	Hop-latency	1 cycle
Routing delay	2/3 cycles	# Virt. channels	4
(w/wo bypassing)		Buffers/port	8
Auxiliary structures (size in number of entries)			
RCB	128	VB	20
MQ	16		

Table 4. Simulation parameters (also see [38])

interconnects (calibrated with Synopsys RSoft [71]). To simulate power and energy consumption, it is bundled with the Orion 2 [34] and McPAT [40] tools. All the optical components such as Y-junctions, waveguide bends, ring resonators, and splitters were simulated using the Synopsis RSoft [71] and Lumerical simulators [43, 44]. We validated the optical parameters and the losses associated with all the devices. Moreover, the electrical components such as the RS, ANN predictor and tuning controller were synthesized in VHDL using the Cadence RTL compiler (at UMC 14 nm technology). Unlike other proposals [53, 73] that use a mean (aggregated) value for the resonator tuning power, we performed detailed thermal simulations of our design, using HotSpot [79]. It was used to estimate the temperature of the die and then calculate the tuning power accordingly. All these tools have been thoroughly verified against real hardware and have been used for evaluating many proposals in this area [7, 54, 57]. We use workloads from the Splash2 [77] and Parsec [12] benchmark suites. Finally, note that we define **performance** as the reciprocal of the simulated execution time.

We compare *PShaRe* with three other state of the art architectures in this field: *ColdBus* [57], *Probe* [82], and *ATAC* [38]. We use the laser modulation and prediction mechanism of *Probe* for a 32-core system and then simulate the system. It predicts optical network usage based on buffer and

Optical parameters	
Wavelength (λ)	$1.55\mu m$
Width of waveguide (W_g)	$3\mu m$
Slab height	$1\mu m$
Rib height	$3\mu m$
Refractive Index of $SiO_2(n_r)$	1.46
Refractive Index of Si (n_c)	3.45
Input Driver Power	76 μW
Insertion Coupling Loss	50%
Photodetector quantum efficiency	0.8 A/W
Photodetector minimum power	36 μW
Combined transmitter and receiver delay	180-270 ps
Optical propagation delay	7 ps/mm
Electrical propagation delay	35 ps/mm
Bending Loss	1 dB
Waveguide Loss	1 dB
Coupler Loss	1 dB
Photodetector	0.1 dB
Off-chip laser Wall Plug Efficiency	30 %
On-chip laser Wall Plug Efficiency	12%
Splitter Loss	0.36 dB
Comb Splitter Loss	0.5 dB
Off Ring Loss	0.001 dB
Ring Modulation	$1\mu W/^\circ C$
Input Photonic heater power	$1mW/^\circ C$
Input Electrical heater power	$34.2\mu W/^\circ C$

Table 5. Optical parameters [15, 20, 47, 52, 53, 55, 57, 62, 73, 76]

link utilization. *ColdBus* [57] is a SWMR crossbar based on-chip photonic network for 32 cores and 32 cache banks. It uses a prediction mechanism based on the PC of memory instructions, along with the past history of hits and misses. To handle mispredictions, it uses a set of extra waveguides that are shared by all the on-chip optical stations. *ATAC* [38] is a kilocore architecture, which uses an unmodulated laser. Based on *ATAC*, we developed a 32 core chip. The chip uses the power delivery network of *ATAC*. It does not use any laser modulation and hence the laser provides a constant amount of power irrespective of the network traffic.

In addition to the above proposals, we also compare the prediction mechanism of *PShaRe* with a linear regression based approach proposed by Winkle et al. [76]. We replaced the ANN predictor in *PShaRe* with their predictor (Section 5.5). A direct comparison of energy and performance is not possible because significant changes need to be made to the DVFS based CPU-GPU design proposed by Winkle et al. [76].

5.1.1 Target System. Let us illustrate a reference design with the ARM Cortex A15 processor. We assume similar cores in our design with some minor differences. Each core has a 4-issue OOO pipeline operating at 2.5GHz. The system has 32 cores and 32 512KB L2 cache banks. Each core has 64KB private data and instruction caches. Using the scaling factors provided by Stan et al. [68], we compute the size of the core to be less than $4mm^2$ at 14nm technology. With such a small core, it is possible to have 32 cores occupying $128mm^2$. We can also integrate 32 cache banks of capacity 16MB on an area less than $60mm^2$ (calculated using Cacti 6.0 and scaled using [68]). **Note that our approach is not specific to the reference design.**

The architectural parameters of the evaluated system are given in Table 4 and the optical parameters of our network are given in Table 5.

5.1.2 Area Evaluation. In terms of hardware cost, *PShaRe* requires some additional optical components to completely realize its objectives. It requires additional hardware for implementing the

Structures	Size per structure	Total Area Overhead
Extra Structures for Prediction		
Gates for multiplications and additions	$146\mu m^2$ per neuron [2, 72]	$\approx 0.06\%$
Lookup table for the Sigmoid function	$780nm^2$ per neuron [46]	$< 10^{-6}\%$
Extra Structures for Sharing		
Ring resonators	$113\mu m^2$ per resonator	0.2%
Waveguide Reservation Table (WRT)	8 Bytes	$10^{-7}\%$
On-chip lasers	$7.68 \times 10^{-3} mm^2$ [76]	
Extra Structures for Power Reuse		
Optical heaters (OH)	$\approx 768\mu m^2$ [27]	$\approx 0.1\%$
Near field transducers (NFT)	$\approx 10000nm^2$ [14]	

Table 6. Area overheads of different structures

power reuse scheme, arbitration scheme and the artificial neural network based predictor. Table 6 shows the individual area of each component. We use an optimized neural network as proposed in [2, 72] that relies on AND, XNOR, and OR gates to perform the additions and multiplications that are required to use a neural network. In addition, it uses a fast look-up table based scheme to implement the non-linear Sigmoid function [46]. For thermal tuning, we designed a standard PID based tuning controller and its area is $\approx 4000\mu m^2$, and its latency is 740 ps. For a detailed discussion on area evaluation, please refer to online Appendix [8]. The additional area required is less than 1.5% of the die size ($260mm^2$).

5.2 Power Models

For the electrical network used for comparison (see Section 5.3), we have assumed state of the art electrical links, requiring 13pJ [9, 33, 53] of energy to send a flit across a link. We require $16pJ$ for a 16-byte flit to traverse the crossbar of a 5×5 electric switch [33, 53]. Thus, a 128-bit flit requires $29pJ/hop/flit$. In *PShaRe* the power consumed by the drivers, ANN predictor, and receivers are the major components of electric power consumption. The ANN performs 48 multiplications and 48 additions, consuming 21pJ [29] of energy. Thus, in an epoch of 100 cycles we require at most $150\mu W$ of power (including both the training and the prediction phases). For all optical power calculations, we have used a standard method (please see [8]). We calculated the energy required for each message by incorporating **all the losses** that occur during the transmission of the message through all the optical components. Using the values from Table 5, we calculated that *PShaRe* requires roughly 1pJ of optical energy to send a single bit.

5.3 Optical Vs Electrical Networks

The notion of replacing an electrical network with an optical network is still a debatable issue. To answer this question we quantify the improvements in performance and energy. Electrical networks are challenged in terms of scalability and bandwidth as observed by [67]. We use a state of the art electrical network proposed by Kim et.al. [37] (please see Table 4 for the parameters of the electrical NoC) and compared it with the proposed optical network. The results of the simulation show that the optical network performs 2.2X better than the electrical network in terms of simulation time (see Figure 5). Others have also reported similar results [53, 73]. As a result, we do not consider electrical networks henceforth. We also reduced the ED^2 by 73%.

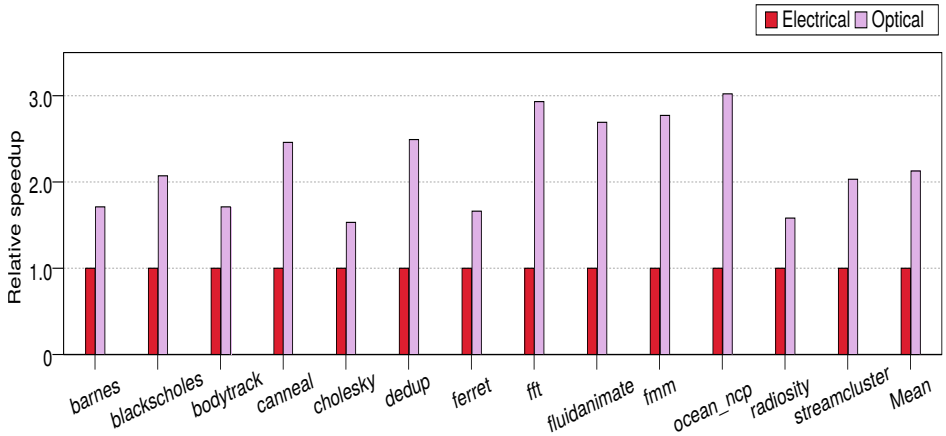

Fig. 5. Optical vs electrical network (performance comparison)

Benchmarks	LASER Prediction Accuracy (%)(A)			Fraction of requests due to reason 1 (B)	Fraction of requests due to reason 2 (C)
	ColdBus Predictor	ML Predictor	ANN Predictor		
barnes	75.4	84.3	97.4	0.072	0.9
blackscholes	91.4	90.6	87.8	0.2	0.78
bodytrack	73.5	75.5	91.8	0.14	0.8
canneal	81.43	85.4	97.56	0.1	0.86
cholesky	85.65	90.3	95.45	0.12	0.831
dedup	77.3	88.2	97.76	0.07	0.91
ferret	80.4	86.5	98.34	0.065	0.93
fft	75.6	87.54	95.7	0.13	0.853
fluidanimate	76.5	84.3	97.6	0.1	0.87
fmm	82.27	90.8	96.38	0.11	0.87
ocean_ncp	72.6	81.4	96.3	0.1	0.875
radiosity	93.33	95.7	90.34	0.18	0.81
streamcluster	72.4	78.9	95.71	0.112	0.86
Mean	79.82	86.2	95.24	0.115	0.858

Table 7. Analysis of prediction accuracy and contention at the stations

5.4 Activity Prediction and Power Consumption

5.4.1 Comparison between Predictors. In this section, the artificial neural network (ANN) based predictor of *PShaRe* is compared with the address based predictor proposed by Peter et al. [57] and the regression based machine learning (ML) approach proposed by Winkle et al. [76]. Column A of Table 7 compares the prediction accuracies of the three different predictors. We want to mention here that in a shared network it is not possibly to ascertain the accurate prediction accuracy. In our results we say that the scheme has correctly predicted if optical power is available to a station if it needs to send a message, and vice versa. Wrong predictions are hurtful both in terms of performance and power consumption. However, our scheme tries to handle both false positives and false negatives by allowing the stations to share the optical power. From Table 7, we observe that the neural network based predictor has 16.2% and 9% more accuracy than the address based and regression based predictors respectively. The higher prediction accuracy of our predictor is responsible for decreasing the static power consumption (See Section 5.4.2) without affecting the performance of the system.

5.4.2 Laser Power Consumption. The total laser power consumed includes the power consumed by off-chip and on-chip lasers. In our scheme, the power generated by on-chip lasers is used by

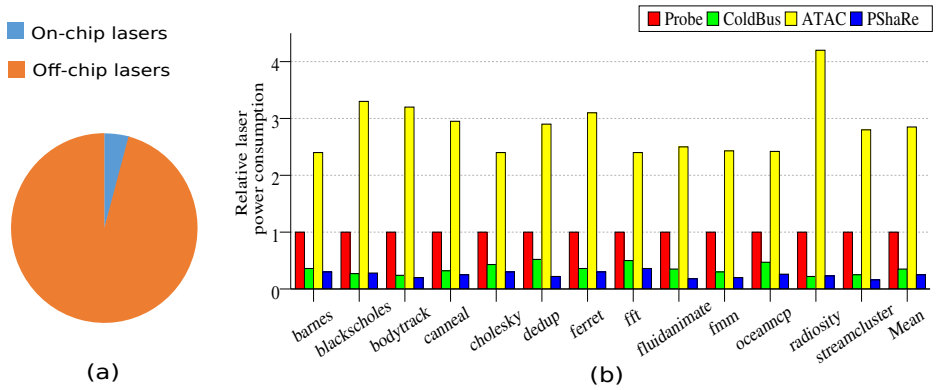

Fig. 6. Laser power comparison

optical stations to communicate with the representative servers, whereas the power generated by off-chip lasers is used by on-chip optical stations to send the data and control messages.

The on-chip lasers generate power on-demand and are used to send a small amount of data. Thus, consuming the least power. However, the power consumed by off-chip lasers depends upon the duration for which the laser is turned on, which in turn directly depends upon the accuracy of the prediction mechanism used to modulate the laser. In Figure 6 (a), we have shown the laser power consumed by on-chip and off-chip lasers. It is clear from the plot that the on-chip lasers consume very small amount of optical power (less than 3.3%), whereas the power consumed by off-chip lasers is the major contributor to the overall power consumption.

In Figure 6 (b), we plot the laser power consumed by different configurations. *ATAC* does not use any laser modulation technique and hence it is the most power consuming configuration. It consumes 3X more power than *Probe*. *PShaRe* consumes the least power because it has a higher prediction accuracy (see Section 5.4.1). Moreover, it allows stations to share the available power, which results in an increase in effective utilization of laser power. It consumes 4X less power as compared to *Probe*. *ColdBus* is the second best configuration, consuming 2.7X less power as compared to *Probe*.

In the case of *blackscholes* and *radiosity*, PShaRe consumes more power as compared to *ColdBus*. It is because in these benchmarks, the address based predictor has an 6% higher prediction accuracy. However, in all other benchmarks, the neural network based predictor performs better than the address based predictor.

5.5 Comparison with ML Based Predictor

In this section, the ANN based predictor of *PShaRe* is compared with the linear regression based predictor proposed by Winkle et al. [76]. We replaced the ANN based predictor in *PShaRe* with the ML based predictor in order to compare the two predictors.

Figure 7(a) shows the relative laser power consumed by *PShaRe* with these two different predictors. It is clear from the figure that the ANN based configuration consumes 13% less power as compared to the design with the ML based predictor. The reason is the higher prediction accuracy of the ANN based predictor. However, in the case of *blackscholes*, and *radiosity*, the ANN based predictor consumes more power because of its lesser prediction accuracy in these two benchmarks. In these two benchmarks, there are more false positives, which resulted in increased power consumption.

Figure 7(b) and Figure 7(c) compare the performance and $energy\_delay^2$ product of the ANN and ML based predictors respectively. The performance of the system directly depends upon the

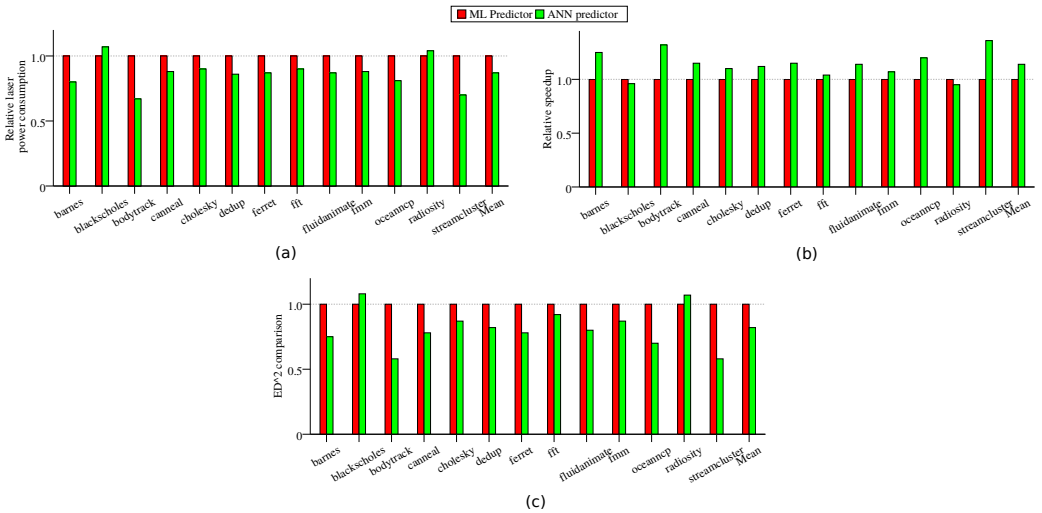

Fig. 7. (a) Relative laser power consumption. (b) Relative speedup. (c) ED^2 comparison

accuracy of predicting the laser power requirement. Since the ANN based predictor has 9% more accuracy in predicting the laser power requirement, we see a 12% increase in performance as compared to the ML based predictor. However, in the *blackscholes* and *radiosity* benchmarks, the ML based predictor performs better because of its higher accuracy in these two benchmarks (along with more false negatives). In the remaining 11 benchmarks, the ANN based predictor has a higher accuracy. In terms of ED^2 values, the ANN based predictor results in a 18% reduction as compared to the ML based predictor.

From these results, it is clear that by using a non-linear predictor, we will be able to predict the laser power requirement better.

5.6 Performance Analysis

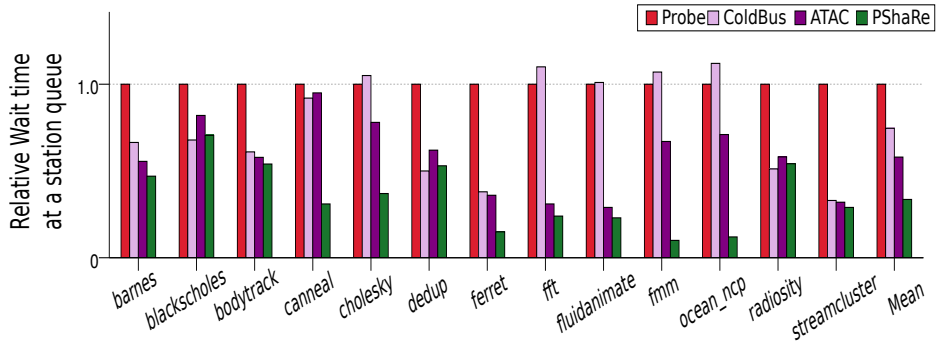

Fig. 8. Relative wait time at a station queue

5.6.1 Contention. Every optical station in any kind of photonic on-chip network has a message queue, which stores message requests from its constituent cores/cache banks. In *PShaRe*, we have assumed a message queue of size 16. The messages sent by a core or a cache bank are stored in the message queue of the respective optical station, and wait for transmission. The message waits in the queue because of three reasons: 1) The laser power is not available (laser off due to mispredictions).

2) Due to high traffic, the station is not able to get access to the power waveguide. 3) The station is busy in sending some other message (only in the case of a single writer based topology).

In *PShaRe*, a message mainly waits because of reason 2 (see Column (B) of Table 7). Very few (roughly 12%) messages wait because of reason 1. Most of the messages (≈ 88%) wait because of high traffic, and high contention in the power waveguide. However, in the case of other configurations (*ColdBus* and *Probe*), all three reasons are seen (more than 11%).

In Figure 8, we plot the relative wait time of a message in a queue for different configurations. *PShaRe* decreases the wait time of messages by nearly 68% as compared to *Probe*, and this is the main reason for improvement in the performance of *PShaRe*. In the case of *fmm*, *ferret* and *ocean_ncp*, *PShaRe* decreases the wait time by 87%, 82%, and 86% respectively, and thus this leads to greater performance improvements in these three workloads (see Section 5.6.2). The main reasons for the reduction in wait time are: *PShaRe* allows stations to send multiple messages at a time; and *PShaRe* is more accurate while predicting the laser power requirement (see Section 5.4.1)

5.6.2 Performance Comparison. Figure 9 shows the relative performance of different configurations. The results are normalized to the *Probe* configuration. In almost all the benchmarks, *PShaRe* performs better than all the other configurations. The main reasons for performance improvement is the decrease in wait time of messages in the message queue (see Section 5.6.1). The better performance of *ATAC* as compared to *Probe* and *ColdBus* is credited to the availability of laser power all the time.

Mispredictions in the case of *Probe* force the stations to wait for the next *epoch*, whereas in *ColdBus* it is handled by allowing a station to use the extra waveguides that carry contingency power. That is why *ColdBus* performs 9% better than *Probe*. In the case of *ferret*, *fmm*, and *ocean_ncp* the greater improvement in performance is clearly attributed to the much higher reduction in contention at the message queues in these three benchmarks. The reason for a slight dip in performance of *PShaRe* in the case of *blackscholes* and *radiosity* is attributed to its lower prediction accuracy in these two cases, which also led to a higher wait time in these two benchmarks. To summarize, it is clear from Figure 9 that *PShaRe* is the best configuration with 31%, 22%, and 15% better mean performance as compared to *Probe*, *ColdBus*, and *ATAC* respectively.

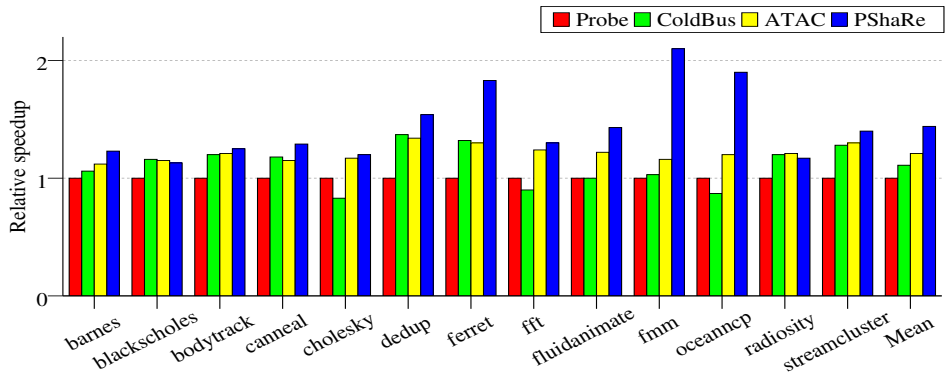

Fig. 9. Performance comparison

5.6.3 ED^2 Comparison. The $Energy\_Delay^2 (ED^2)$ product is the standard metric used to compare full systems. Note that here the energy represents the full system's energy including the cores, caches and NoC. Figure 10 shows the ED^2 comparison between different configurations. From the figure, it is clear that *PShaRe* is the best configuration with a 39% reduction in ED^2 as compared to *Probe*. *ATAC* has the highest ED^2, which is 47% more than *PShaRe*. The reason is its unmodulated

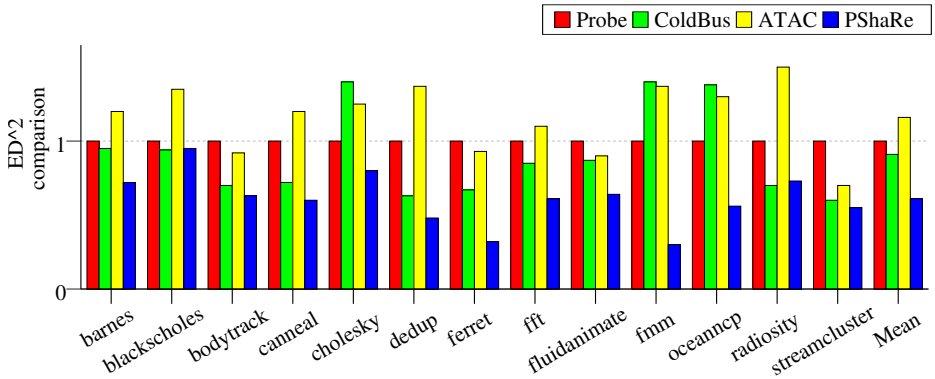

Fig. 10. ED^2 comparison

laser that supplies a constant amount of power irrespective of network requirements. As compared to *ColdBus*, *PShaRe* results in a 33% reduction in ED^2.

5.7 Power breakdown

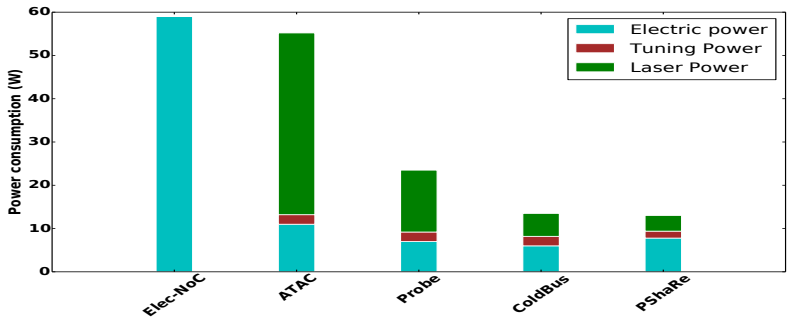

Fig. 11. Network power consumption breakdown

In this section, we present the overall power consumption breakdown in various on-chip networks (see Figure 11). An electrical network (ENOC) consumes only electrical power and this is much more than any of the photonic on-chip networks. The reason is that in an electrical network the message has to pass through electrical links and electrical routers which result in elevated power consumption. In photonic NoCs (PNOCs), the proportion of electrical energy is far lower (15%). However, from Figure 11 we observe that among different PNOCs, *ATAC* has the highest electrical power consumption followed by *PShaRe*. The reason is that *ATAC* uses a separate electrical control network for sending control messages. In the case of *PShaRe* the higher electrical power consumption is attributed to its prediction and arbitration schemes. The tuning power is nearly the same in all the PNOCs other than *PShaRe* since they use almost an equal number of ring resonators. However, due to our power reuse scheme, *PShaRe* reduces the tuning power by roughly 38%.

5.8 Comparison between the Arbitration Schemes

In this section, we compare the laser energy consumed by our novel arbitration scheme with the conventional token based arbitration scheme used in [57]. In the token based arbitration scheme, the tokens (optical pulses) continuously circulate inside the token waveguide, irrespective of their usage. The number of tokens inside the token waveguide directly depends upon the number of stations

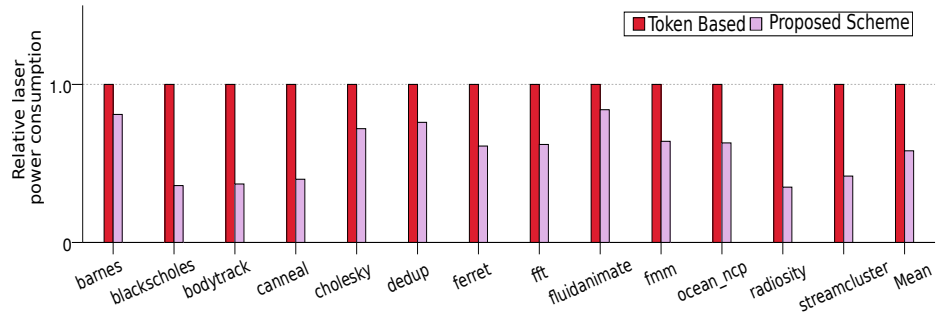

Fig. 12. Relative laser power consumed by different arbitration schemes

predicted to transmit in the current epoch. Moreover, in [57], we also broadcast a reservation flit, informing the receivers that they need to start receiving data. However, in the case of our arbitration scheme, a short pulse is directly sent from a station to its representative server and in response it gets a short message containing its access rights. There is thus no need to broadcast a reservation flit. Instead a direct message is sent to the receivers by their respective RSs informing them about the arrival of data. The optical power required for the purpose of arbitration and reservation in *PShaRe* is generated by small on-chip lasers. Figure 12 shows the relative laser power consumed by the two schemes. We observe that our scheme consumes 42% less power as compared to the token based scheme.

5.9 Effect of Different Epoch Sizes

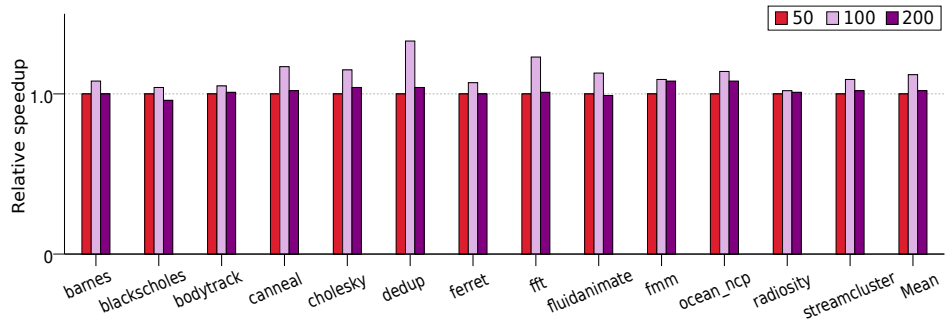

Fig. 13. Performance comparison using different epoch sizes

The choice of the *epoch* size affects the performance and the laser power consumed by the system. In every *epoch*, we have some cycles in which the network is inactive(1-3 cycles). A small *epoch* size results in a relative increase in the total number of inactive cycles. However, using a large *epoch* size implies that we are sending a constant amount of light for a large amount of time, and thus the system is unresponsive. Additionally, a large epoch size results in the fetch and execution of memory instructions in the same cycle, which makes the prediction mechanism fairly ineffective. As a result, we should have a moderate *epoch* size such that the network inactivity time is negligible as compared to the *epoch* size. Also, the epoch size should be such that most of the memory instructions are fetched and executed in different epochs. We choose three epoch sizes that satisfy these criteria – 50,100, and 200 – and compared the performance of *PShaRe* for these three epoch sizes. Figure 13 shows the relative performance of *PShaRe* for these three epoch sizes. From the graph it is clear that *PShaRe* performs 12% better for an epoch of size 100 as compared to its nearest competitor (50-cycle epoch). As a result, we use a 100-cycle epoch in our simulations.

5.10 Reuse of Wasted Power

Benchmark	Temperature range($^\circ C$)	Unused power (W)	Reduction in tuning power (%)
barnes	45-71	3.6	43
blackscholes	45-65	2.6	34
bodytrack	45-63	2.7	37
canneal	45-61	3.1	38
cholesky	45-73	3.9	49
dedup	45-56	2.3	33
ferret	45-60	2.45	34
fft	45-76	3.7	44
fluidanimate	45-70	2.6	38
fmm	45-72	3.8	45
ocean_ncp	45-75	3.3	41
radiosity	45-70	2.3	29
streamcluster	45-60	2.34	32
Mean	45-68	2.9	38.2

Table 8. Thermal simulations and unused power

In addition to using HotSpot at the chip level, we use a finite element CFD tool, COMSOL (for simulating the response of the controller at the micron scale). We created a layout of the chip that includes both the electrical and optical components. The layout along with the periodic power profile (including both electrical and optical power) obtained for a benchmark was supplied to HotSpot. HotSpot gave us the temperature profile of the chip. This procedure was conducted for every benchmark and the thermal variations were calculated accordingly (see Table 8).

As we can see from Table 8, the maximum temperature that the chip reaches is $76^\circ C$. We tune the microring resonators to this maximum temperature, plus a small margin (to $85^\circ C$). The ring resonators are designed to work at this maximum temperature. The amount of heat required to tune the ring resonators is calculated based on this maximum temperature. We have roughly 50,000 ring resonators and we need $1\mu W/^\circ C$ [26, 52, 53, 75] to tune each resonator. Some portion of this heat is generated using the unused optical power (as mentioned in Section 4.4.2) and the remaining heat is generated using traditional electrical micro heaters. On an average, nearly 38% of tuning (see Table 8) is done using the unused optical power and the remaining heat is generated using the traditional micro heaters. Our PID controller ensures that the temperature remains within 0.1°C of the target (acceptable as per [61]).

6 CONCLUSION

Prohibitive static power consumption is a major issue in on-chip optical networks. To predict the network traffic in optical networks and subsequently reduce the unused laser power, we observe that a non-linear predictor such as a neural network is required. It is 9-10% more accurate than state of the art predictors that use regression based machine learning. Moreover, it is necessary to share the available photonic channels in order to reduce their number and increase their utilization. This is done intelligently using a combination of on-chip and off-chip lasers to further reduce static power consumption in an altruistic fashion. Finally, instead of wasting optical power and dissipating it as heat, we propose to use it for productive purposes such as tuning the ring resonators, within limits placed by thermodynamics. By using these techniques, we reduce the laser power by 4.7X as compared to the nearest state of the art proposal. Moreover, we perform 31% better in terms of execution time and decrease the ED^2 by 39% as compared to other state of the art proposals.

REFERENCES

[1] Challener W. A., Peng Chubing, Itagi A. V., Karns D., Peng Wei, Peng Yingguo, XiaoMin Yang, Xiaobin Zhu, N. J. Gokemeijer, Y.-T. Hsia, G. Ju, Robert E. Rottmayer, Michael A. Seigler, and E. C. Gage. 2009. Heat-assisted magnetic recording by a near-field transducer with efficient optical energy transfer. *Nature Photonics* (4 2009).

[2] A. Ardakani, F. Leduc-Primeau, N. Onizawa, T. Hanyu, and W. J. Gross. 2017. VLSI Implementation of Deep Neural Network Using Integral Stochastic Computing. *IEEE Transactions on Very Large Scale Integration (VLSI) Systems* (Oct 2017).

[3] NEW ATLAS. 2017. HP unleashes The Machine memory-centric supercomputer prototype. *https://newatlas.com/hewlett-packard-enterprise-the-machine-big-data/49561/* (2017).

[4] G. Baffou, R. Quidant, and C. Girard. 2009. Heat generation in plasmonic nanostructures: Influence of morphology. *Applied Phys. Letters* (March 2009).

[5] Janibul Bashir, Eldhose Peter, and Smruti R. Sarangi. 2019. BigBus: A Scalable Optical Interconnect. *J. Emerg. Technol. Comput. Syst.* 15, 1 (jan 2019).

[6] Janibul Bashir, Eldhose Peter, and Smruti R. Sarangi. 2019. A Survey of On-Chip Optical Interconnects. *ACM Comput. Surv.* 51, 6 (jan 2019).

[7] Janibul Bashir and Smruti R Sarangi. 2017. NUPLet: A Photonics Based Multi-Chip NUCA Architecture. In *2017 IEEE 35th International Conference on Computer Design (ICCD)*. IEEE.

[8] Janibul Bashir and Smruti Ranjan Sarangi. 2019. Online Appendix: Predict, Share, and Recycle your Way to Low Power Nanophotonic networks. http://www.cse.iitd.ac.in/~srsarangi/tr/pshare-appendix.pdf

[9] Christopher Batten, Ajay Joshi, Jason Orcutt, Anatoly Khilo, Benjamin Moss, Charles Holzwarth, Milos Popovic, Hanqing Li, Henry I Smith, Judy Hoyt, and F. X. Kartner. 2008. Building manycore processor-to-dram networks with monolithic silicon photonics. In *High Performance Interconnects (HOTI)*. IEEE, 21–30.

[10] George B.P. Bezerra, Stephanie Forrest, Melanie Moses, Al Davis, and Payman Zarkesh-Ha. 2010. Modeling NoC Traffic Locality and Energy Consumption with Rent's Communication Probability Distribution. In *Proceedings of the 12th ACM/IEEE International Workshop on System Level Interconnect Prediction (SLIP '10)*. 3–8.

[11] Samarth Bhargava. 2015. *Heat-assisted magnetic recording: Fundamental limits to inverse electromagnetic design*. University of California, Berkeley.

[12] C. Bienia, S. Kumar, J. P. Singh, and K. Li. 2008. The PARSEC benchmark suite: characterization and architectural implications. In *PACT*.

[13] Rodolfo E Camacho-Aguilera, Yan Cai, Neil Patel, Jonathan T Bessette, Marco Romagnoli, Lionel C Kimerling, and Jurgen Michel. 2012. An electrically pumped germanium laser. *Optics express* 20, 10 (2012), 11316–11320.

[14] William A Challener, Ed Gage, Amit Itagi, and Chubing Peng. 2006. Optical transducers for near field recording. *Japanese journal of applied physics* (2006).

[15] Zachary J Coppens, Wei Li, D Greg Walker, and Jason G Valentine. 2013. Probing and controlling photothermal heat generation in plasmonic nanostructures. *Nano letters* 13, 3 (2013).

[16] Y. Demir and N. Hardavellas. 2016. SLaC: Stage laser control for a flattened butterfly network. In *2016 IEEE International Symposium on High Performance Computer Architecture (HPCA)*.

[17] Christopher T. DeRose, Michael R. Watts, Douglas C. Trotter, David L. Luck, Gregory N. Nielson, and Ralph W. Young. 2010. Silicon Microring Modulator with Integrated Heater and Temperature Sensor for Thermal Control. *Conference on Lasers and Electro-Optics 2010*.

[18] John E. Bowers Di Liang. 2010. Recent progress in lasers on silicon. *Nat Photon* (August 2010).

[19] Alexander W Fang, Hyundai Park, Oded Cohen, Richard Jones, Mario J Paniccia, and John E Bowers. 2006. Electrically pumped hybrid AlGaInAs-silicon evanescent laser. *Optics express* 14, 20 (2006), 9203–9210.

[20] Q. Fang, J. Song, X. Luo, L. Jia, M. Yu, G. Lo, and Y. Liu. 2012. High Efficiency Ring-Resonator Filter With NiSi Heater. *IEEE Photonics Technology Letters* (March 2012).

[21] M. FAUGERON, M. Chtioui, A Enard, O. Parillaud, F. Lelarge, M. Achouche, J. Jacquet, A Marceaux, and F. van Dijk. 2013. High Optical Power, High Gain and High Dynamic Range Directly Modulated Optical Link. *Lightwave Technology, Journal of* 31, 8 (April 2013), 1227–1233.

[22] Mickaël Faugeron, Michaël Tran, François Lelarge, Mourad Chtioui, Yannick Robert, Eric Vinet, Alain Enard, Joël Jacquet, and Frederic Van Dijk. 2012. High-Power, Low RIN 1.55-Directly Modulated DFB Lasers for Analog Signal Transmission. *Photonics Technology Letters* 24, 2 (2012), 116–118.

[23] Rajib R Ghosh, Janib Bashir, Smruti R Sarangi, and Anuj Dhawan. 2019. SpliESR: Tunable Power Splitter Based on an Electro-Optic Slotted Ring Resonator. *Optics Communications* (2019).

[24] Alexander O Govorov and Hugh H Richardson. 2007. Generating heat with metal nanoparticles. *Nano today* 2, 1 (2007), 30–38.

[25] Martin T Hagan, Howard B Demuth, Mark H Beale, and Jesus O D. 1996. *Neural network design*. Vol. 20. Pws Pub. Boston.

[26] P. K. Hamedani, N. E. Jerger, and S. Hessabi. 2014. QuT: A low-power optical Network-on-Chip. In *2014 Eighth IEEE/ACM International Symposium on Networks-on-Chip (NoCS)*.
[27] Jiaming Hao, Lei Zhou, and Min Qiu. 2011. Nearly total absorption of light and heat generation by plasmonic metamaterials. *Phys. Rev. B* (Apr 2011).
[28] Robert Hecht-Nielsen. 1992. Theory of the backpropagation neural network. In *Neural networks for perception*. Elsevier.
[29] M. Horowitz. 2014. 1.1 Computing's energy problem (and what we can do about it). In *2014 IEEE International Solid-State Circuits Conference Digest of Technical Papers (ISSCC)*.
[30] Wei Huang, Mircea R Stan, Karthik Sankaranarayanan, Robert J Ribando, and Kevin Skadron. 2008. Many-core design from a thermal perspective. In *Design Automation Conference, 2008. DAC 2008. 45th ACM/IEEE*. IEEE, 746–749.
[31] M. J. Humphrey. 1994. *Calculation of coupling between tapered fiber modes and whispering-gallery modes of a spherical microlaser*. Ph.D. Dissertation. University of Maryland, College Park, Maryland.
[32] Intel. 2013. PCI Express Architecture. *https://itpeernetwork.intel.com/fujitsu-lights-up-pci-express-with-intel-silicon-photonics/* (2013).
[33] Ajay Joshi, Christopher Batten, Yong-Jin Kwon, Scott Beamer, Imran Shamim, Krste Asanovic, and Vladimir Stojanovic. 2009. Silicon-photonic clos networks for global on-chip communication. In *NoCS*.
[34] Andrew B Kahng, Bin Li, Li-Shiuan Peh, and Kambiz Samadi. 2011. Orion 2.0: A power-area simulator for interconnection networks. IEEE.
[35] Ahmad Khonsari, Mohammad R Aghajani, Arash Tavakkol, and Mohammad S Talebi. 2008. Mathematical analysis of buffer sizing for network-on-chips under multimedia traffic. In *Computer Design, 2008. ICCD 2008. IEEE International Conference on*. IEEE, 150–155.
[36] Gun-Duk Kim, Hak-Soon Lee, Chang-Hyun Park, Sang-Shin Lee, Boo Tak Lim, Hee Kyoung Bae, and Wan-Gyu Lee. 2010. Silicon photonic temperature sensor employing a ring resonator manufactured using a standard CMOS process. *Opt. Express* 18, 21 (Oct 2010), 22215–22221.
[37] John Kim, James Balfour, and William Dally. 2007. Flattened butterfly topology for on-chip networks. In *Micro*. 172–182.
[38] George Kurian, Jason E Miller, James Psota, Jonathan Eastep, Jifeng Liu, Jurgen Michel, Lionel C Kimerling, and Anant Agarwal. 2010. ATAC: a 1000-core cache-coherent processor with on-chip optical network. In *PACT*.
[39] Jacob S Levy, Yoshitomo Okawachi, Michal Lipson, Alexander L Gaeta, and Kasturi Saha. 2011. High-performance silicon-based multiple wavelength source. In *CLEO: Science and Innovations*. OSA, CMAA7.
[40] Sheng Li, Jung Ho Ahn, Richard D Strong, Jay B Brockman, Dean M Tullsen, and Norman P Jouppi. 2009. McPAT: an integrated power, area, and timing modeling framework for multicore and manycore architectures. In *MICRO*.
[41] Dong-Kwon Lim, Aoune Barhoumi, Ryan G Wylie, Gally Reznor, Robert S Langer, and Daniel S Kohane. 2013. Enhanced photothermal effect of plasmonic nanoparticles coated with reduced graphene oxide. *Nano letters* 13, 9 (2013), 4075–4079.
[42] Jifeng Liu, Xiaochen Sun, Rodolfo Camacho-Aguilera, Lionel C Kimerling, and Jurgen Michel. 2010. Ge-on-Si laser operating at room temperature. *Optics letters* 35, 5 (2010), 679–681.
[43] Lumerical. 2003. Lumerical DEVICE. (2003). https://https://www.lumerical.com/tcadproducts/device/
[44] Lumerical. 2003. Lumerical mode sol. (2003). https://https://www.lumerical.com/tcadproducts/mode/
[45] LUXTERA. 2001. LUXTERA: Fibre to the Chip. *http://www.luxtera.com/luxtera/products* (2001).
[46] P. Kumar Meher. 2010. An optimized lookup-table for the evaluation of sigmoid function for artificial neural networks. In *2010 18th IEEE/IFIP International Conference on VLSI and System-on-Chip*.
[47] Randy Morris, Evan Jolley, and Avinash Karanth Kodi. 2014. Extending the performance and energy-efficiency of shared memory multicores with nanophotonic technology. *Parallel and Distributed Systems, IEEE Transactions on* 25, 1 (2014), 83–92.
[48] Berkeley News. 2015. Engineers demo first processor that uses light for ultrafast communications. *http://news.berkeley.edu/2015/12/23/electronic-photonic-microprocessor-chip/* (2015).
[49] C. Nitta, M. Farrens, and V. Akella. 2011. Addressing system-level trimming issues in on-chip nanophotonic networks. In *HPCA*. 122–131.
[50] G Oskam, JG Long, A Natarajan, and PC Searson. 1998. Electrochemical deposition of metals onto silicon. *Journal of Physics D: Applied Physics* 31, 16 (1998), 1927.
[51] Kishore Padmaraju and Keren Bergman. 2014. Resolving the thermal challenges for silicon microring resonator devices. *Nanophotonics* 3, 4-5 (2014), 269–281.
[52] Yan Pan, John Kim, and Gokhan Memik. 2010. Flexishare: Channel sharing for an energy-efficient nanophotonic crossbar. In *HPCA*.
[53] Yan Pan, Prabhat Kumar, John Kim, Gokhan Memik, Yu Zhang, and Alok Choudhary. 2009. Firefly: illuminating future network-on-chip with nanophotonics. In *ACM SIGARCH Computer Architecture News*. ACM.
[54] Eldhose Peter, Anuj Arora, Janibul Bashir, Akriti Bagaria, and Smruti R. Sarangi. 2017. Optical Overlay NUCA: A High-Speed Substrate for Shared L2 Caches. *J. Emerg. Technol. Comput. Syst.* 13, 4, Article 53 (2017). https:

//doi.org/10.1145/3064833
[55] Eldhose Peter and Smruti R Sarangi. 2014. OptiKit: An Open Source Kit for Simulation of On-Chip Optical Components. (2014).
[56] Eldhose Peter and Smruti R Sarangi. 2015. Optimal Power Efficient Photonic SWMR Buses. In *Silicon Photonics (with HiPEAC)*.
[57] Eldhose Peter, Arun Thomas, Anuj Dhawan, and Smruti R Sarangi. 2015. ColdBus: A Near-Optimal Power Efficient Optical Bus. In *HiPC*.
[58] Intel Silicon Photonics. 2017. Moving Data With Light. *https://www.intel.in/content/www/in/en/architecture-and-technology/silicon-photonics/silicon-photonics-overview.html* (2017).
[59] Andrew W Poon, Fang Xu, and Xianshu Luo. 2008. Cascaded active silicon microresonator array cross-connect circuits for WDM networks-on-chip. In *Integrated Optoelectronic Devices 2008*. International Society for Optics and Photonics, 689812–689812.
[60] Circuits Multi Projects. 2018. Silicon Photonics. *https://mycmp.fr/datasheet/silicon-photonic-ics-si310-phmp2m* (2018).
[61] Ciyuan Qiu, Jie Shu, Zheng Li, Xuezhi Zhang, and Qianfan Xu. 2011. Wavelength tracking with thermally controlled silicon resonators. *Optics Express* 19, 6 (2011), 5143–5148.
[62] Graham T. Reed. 2008. *Silicon Photonics: The State of the Art*. John Wiley & Sons.
[63] IBM Research. 2012. Silicon Photonics. *https://www.zurich.ibm.com/st/photonics/wdm.html* (2012).
[64] MIT Technology Review. 2008. A Record-Breaking Optical Chip. *https://www.technologyreview.com/s/410383/a-record-breaking-optical-chip/* (2008).
[65] Atsushi Sakai, Tatsuhiko Fukazawa, and Toshihiko Baba. 2002. Low loss ultra-small branches in a silicon photonic wire waveguide. *IEICE Transactions on Electronics* (2002).
[66] S. R. Sarangi, Kalayappan Rajshekar, Kallurkar Prathmesh, Goel Seep, and Peter Eldhose. 2015. Tejas: A Java based Versatile Micro-architectural Simulator,. In *PATMOS*.
[67] Md Ashif I Sikder, Avinash K Kodi, Matthew Kennedy, Savas Kaya, and Ahmed Louri. 2015. OWN: Optical and Wireless Network-on-Chip for Kilo-core Architectures. In *High-Performance Interconnects (HOTI)*. 44–51.
[68] Mircea R. Stan, Kevin Skadron, Wei Huang, and Karthick Rajamani. 2011. Scaling with Design Constraints: Predicting the Future of Big Chips. *IEEE Micro* 31 (2011).
[69] STH. 2019. Intel Silicon Photonics Update at Interconnect Day 2019. *https://www.servethehome.com/intel-silicon-photonics-update-at-interconnect-day-2019/* (2019).
[70] C. Sun, M. T. Wade, Y. Lee, J. S. Orcutt, L. Alloatti, M. S. Georgas, A. S. Waterman, J. M. Shainline, R. R. Avizienis, S. Lin, and B. R. MOSS. 2015. Single-chip microprocessor that communicates directly using light. *Nature* (2015).
[71] SYNOPSYS. 1990. SYNOPSYS RSoft. (1990). https://www.synopsys.com/optical-solutions/rsoft.html.
[72] E. Vahapoglu and M. Altun. 2016. Accurate Synthesis of Arithmetic Operations with Stochastic Logic. In *2016 IEEE Computer Society Annual Symposium on VLSI (ISVLSI)*.
[73] Dana Vantrease, Robert Schreiber, Matteo Monchiero, Moray McLaren, Norman P. Jouppi, Marco Fiorentino, Al Davis, Nathan Binkert, Raymond G. Beausoleil, and Jung Ho Ahn. 2008. Corona: System Implications of Emerging Nanophotonic Technology. In *ISCA*.
[74] Jing Wang, Yiting Chen, Xi Chen, Jiaming Hao, Min Yan, and Min Qiu. 2011. Photothermal reshaping of gold nanoparticles in a plasmonic absorber. *Opt. Express* (Jul 2011).
[75] S. Werner, J. Navaridas, and M. LujÃąn. 2015. Amon: An Advanced Mesh-like Optical NoC. In *2015 IEEE 23rd Annual Symposium on High-Performance Interconnects*.
[76] S. Van Winkle, A. K. Kodi, R. Bunescu, and A. Louri. 2018. Extending the Power-Efficiency and Performance of Photonic Interconnects for Heterogeneous Multicores with Machine Learning. In *2018 IEEE International Symposium on High Performance Computer Architecture (HPCA)*.
[77] Steven Cameron Woo, Moriyoshi Ohara, Evan Torrie, Jaswinder Pal Singh, and Anoop Gupta. 1995. The SPLASH-2 programs: characterization and methodological considerations. *SIGARCH Comput. Archit. News* 23 (May 1995), 24–36.
[78] Qianfan Xu, Bradley Schmidt, Sameer Pradhan, and Michal Lipson. 2005. Micrometre-scale silicon electro-optic modulator. *Nature* (May 2005).
[79] R. Zhang, M. R. Stan, and K. Skadron. 2015. Hotspot 6.0: Validation, acceleration and extension. (2015).
[80] Xiang Zhang and Ahmed Louri. 2010. A multilayer nanophotonic interconnection network for on-chip many-core communications. In *Proceedings of the 47th Design Automation Conference*. ACM, 156–161.
[81] Jiwen Zheng, Zihua Zhu, Haifeng Chen, and Zhongfan Liu. 2000. Nanopatterned assembling of colloidal gold nanoparticles on silicon. *Langmuir* 16, 10 (2000), 4409–4412.
[82] Li Zhou and Avinash Karanth Kodi. 2013. Probe: Prediction-based optical bandwidth scaling for energy-efficient nocs. In *NOCS*.