Fast Exploration of Silicon Photonic Network Designs for Exascale Systems

Sebastien Rumley, Robert Hendry, Keren Bergman
Lightwave Research Lab, Columbia University, New York, NY
sr3061@columbia.edu, rh2519@columbia.edu, bergman@ee.columbia.edu

***Abstract*—An approach for exploring the potential applications and performance-energy benefits of silicon photonic technology in future computing systems with a particular focus on Exascale design considerations is presented.**

I. INTRODUCTION

The realization of Exascale systems is scheduled to be reached in about five years [1]. However, neither traditional transistor scaling nor the recent trend in increased hardware parallelism will be sufficient for achieving an Exascale design within the power and performance design constraints, particularly due to the challenges related to data movement [2]. Silicon photonic (SiP) technology has become a promising candidate for the Exascale system's communication medium, providing very high bandwidth density and low power compared to electronic solutions. SiP technologies have been demonstrated to achieve optical switching at nanosecond speeds and unlike other optical technologies, this photonic platform can be incorporated on the same silicon die, i.e. very close to the computing resources.

SiP links and networks, however, are not one-to-one replacements for electronic interconnects. The specific functionalities, namely dense WDM, buffer-less operation, long-haul communication, and potentially even optical signal processing, fundamentally change the way computer networks must be built and operated. As a result, the integration of silicon photonics into computing systems requires system-wide redesign. Moreover, to obtain the characteristics that make this technology so appealing, SiP devices need to be carefully engineered for any particular network design. This interdependence requires a comprehensive design flow where device and network designs are developed simultaneously. At the same time, the specificity of Exascale systems might oblige programmers to write tailored software, with particular requirements in terms of data movement. There is therefore also an interdependence between the application and the network, and consequently between the SiP devices and the application.

These requirements call for a set of modeling and simulation tools to realize the development and design exploration of SiP network architectures. In this paper we present a software environment called PhoenixSim 2.0, enabling the design exploration and performance modeling of new SiP enabled networks for Exascale applications.

II. ENVIRONMENT DESCRIPTION

From a methodological point of view, the PhoenixSim 2.0 environment focuses on the early exploration of the immense design space offered by integrated silicon photonics. For the design of the interconnect within Exascale systems. the initial work involves the identification of the physically possible zone of interests inside the candidate architectures design space. We consider the application performance and the feasibility as the strategic metrics of interest. Within the design environment, an architecture is declared infeasible either because it does not guarantee the data integrity (due to the optical impairments) or due to excessive power consumption.

To obtain estimates in terms of feasibility and application performance, PhoenixSim 2.0 orchestrates three independent tools. The first one is a physical layer model (Photonic Interconnect Link Optimisation Tool - PILOT) that analyses the impact of each SiP device on the signal quality. PILOT also maximizes the number of wavelengths used on a link (and by this way the realized bandwidth) while ensuring signal integrity. Insertion losses, noise figures and power penalties are calculated for each device. In addition to including detailed device parameters, such as coupling coefficients, PILOT calculates each type of loss as a function of the wavelength channel spacing, yielding an accurate prediction of the link maximum feasible bandwidth density. PILOT integrates device models reported in the literature and laboratory measurements, as well as projections based on these models. PILOT is developed to support several symbol rate and/or modulation formats.

The second tool, LWSim (LightWeight SIMulator), is a discrete event simulator which analyses the statistical behavior of the network of interest in presence of application-like traffic. It intends to understand how the network topology and the resource reservation/allocation mechanisms impair the theoretical bandwidth and latency. The optical network models developed within LWSim integrate timing measurements obtained in the laboratory or reported in the literature. LWSim can inject random traffic into the network model, or run application skeletons [3], which capture traffic dependences between network clients.

The third tool, Javanco [4], provides the data structures and associated API to create and handle network descriptions that are then passed to either LWSIM or PILOT. The API also regroups graph theory related functionalities than can be used to reason about the network itself. By this mean, general metrics as the bisectional bandwidth or minimal latency can quickly be obtained. Javanco finally provides graphical and visualization features can that be leveraged for debugging, verification, and reporting.

We end up with the following design flow: a baseline network topology is first created within Javanco. It is then sent to PILOT where the highest supported bandwidth without compromising signal quality is determined. Parameters optimized by PILOT are integrated in the description, which is then passed to LWSim. Simulations are then run to evaluate how the topology performs under arbitrary traffic (random or skeleton applications based). Upon analysing the results, the baseline topology can be modified toward achieving better application performance. *Because the tools share a common data structure, we can iteratively make changes to the abstract design and quickly determine how these changes in network topology impact both performance and signal quality.*

Existing simulation environments often collapse time-dependent (i.e. depending on the network state) and time-independent calculations into monolithic experiments. In our opinion, combining these calculations (e.g. studying network congestion during the execution of application while measuring the optical attenuation in a waveguide) encourages ad-hoc and inextensible models, and generally reduces the environment flexibility while increases simulation execution time.

In contrast, the PhoenixSim 2.0 environment is broken into discrete tools in a modular fashion. This provides the ability to run faster, more comprehensive simulations, and provides greater flexibility in how we combine our timing models and our static physical models. Having well defined tools helps also a lot to maintain and extend each one, depending on the needs, without having to disrupt the whole structure. We achieve this modularity without sacrificing the organization of a monolithic simulation via our common network description in Javanco. As a result, PhoenixSim 2.0 is an agile environment that spans from SiP to applications, over which designers can undertake various experiments for validating or invalidating candidate architectures.

III. Related Work

Researchers in the field have studied the integration of silicon photonics into computing systems with various strategies. DSENT [5] takes a CMOS-integration approach and focuses on the circuitry required to drive the optical lanes. This provides accurate estimations of the system power and area consumption, however, only under fixed network load. Other groups focus on integrating optical network models with the generic hardware system simulation, using SystemC, which allows functional verification as well as extended timing measurements. Models of optical networks have also been incorporated in cycle-accurate system wide simulators [6, 7].

These approaches result in major advances in SiP enabled network understanding, and are necessary for validating well-developed network designs. They however have their own fields and metrics of interest, and neglect other effects. For example, DSENT doesn't consider traffic variations, while cycle-accurate simulators do not consider the optical signal quality. Among these tools, to ones fitting to our approach (concise description as input, fast execution) could potentially be integrated in our environment. In particular, DSENT could be used to evaluate the feasibility of a design from driving circuitry point of view. The other tools, i.e. the ones to complex to put in operation to offer fast exploration, can be used to further study and optimize the candidates retained in the first exploration round.

This is what we intend to achieve through our collaboration with the Sandia National Labs. Their simulator, SST-Macro [3, 6], is a coarse grained simulator able to simulate parallel applications written in various standards (MPI, OpenSHMEM, etc.) on a very large number of clients (>1000), a must for Exascale modeling. After candidate designs are selected by PhoenixSim 2.0, we will implement detailed network models in SST-Macro and utilize its extensive application model library to refine our designs under realistic application traffic. Beyond or in parallel to these analyses, network models of interest could also be transformed into SST-Micro `components` [7], and simulated/validated in a cycle accurate simulation environment including detailed processor and memory models as gem5 and DRAMSim [8].

IV. Environment Development Progress

The considered approach aims at integrating as close as possible application performance measurements and optical signal considerations while maintaining a flexible, extensible, and fast-to-configure and -to-execute software environment. The physical-layer analysis approach in PILOT has previously been implemented in ad-hoc tools, and is currently being developed into software satisfying these criteria. LWSim currently implements a small amount of network component models and network protocols. We are currently adding more models, in particular traffic models that mimic important applications. We are also looking at integrating more tools, targeting the schematics of Figure 1.

V. Conclusion

Due to the revolutionary nature of integrated silicon photonics, the design space for Exascale network architectures is very large and vastly unexplored. When designing silicon photonic networks, however, care must be taken in order to ensure the integrity of optically modulated data and physical link layers. The presented software environment quickly reports both the expected performance improvement and a feasibility analysis of any target SiP network architectures.

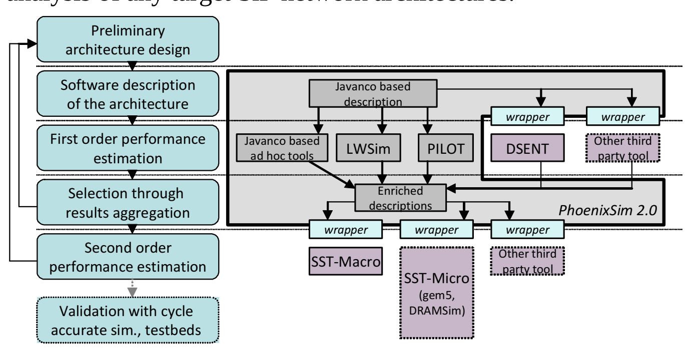

Figure 1. Figure 1: Schematics of our targeted design flow

References

[1] www.top500.org

[2] J. Shalf, *et al*. "Exascale Computing Technology Challenges," *VECPAR*, 2010

[3] C. L. Janssen, *et al*. "A Simulator for Large-Scale Parallel Computer Architectures," *(IJDST)* 1, 2010.

[4] S. Rumley *et al*. "*Software Tools and Methods for Research and Education in Optical Networks*." Springer, 2009.

[5] C. Sun, *et al.*, "DSENT -- A Tool Connecting Emerging Photonics with Electronics for Opto-Electronic (NOCS), 2012.

[6] sst.sandia.gov

[7] A. F. Rodrigues, *et al*. "The structural simulation toolkit". *SIGMETRICS Perform. Eval. Rev.* 38, 4 (March 2011)

[8] M. Hsieh, *et al*. "SST + gem5 = a scalable simulation infrastructure for high performance computing". SIMUTOOLS '12.