OPTISIM: A SYSTEM SIMULATION METHODOLOGY FOR OPTICALLY INTERCONNECTED HPC SYSTEMS

ALTHOUGH CAD TOOLS HAVE SIGNIFICANTLY ASSISTED ELECTRONIC SYSTEM SIMULATION, THE SYSTEM-LEVEL OPTOELECTRONICS MODELING FIELD HAS LAGGED BEHIND DUE TO A LACK OF SIMULATION METHODOLOGIES AND TOOLS. OPTISIM, A SYSTEM-LEVEL MODELING AND SIMULATION METHODOLOGY OF OPTICAL INTERCONNECTS FOR HPC SYSTEMS, CAN PROVIDE COMPUTER ARCHITECTS, DESIGNERS, AND RESEARCHERS WITH A HIGHLY OPTIMIZED, EFFICIENT, AND ACCURATE DISCRETE-EVENT ENVIRONMENT TO TEST VARIOUS HPC SYSTEMS.

Avinash Karanth Kodi
Ohio University

Ahmed Louri
University of Arizona

The computer and communication industries have recognized that short-range (chip-to-chip and board-to-board) optical interconnects could potentially provide a cost-effective solution to the increasing communication bandwidth demands of high-performance computing (HPC) systems.[1,2] Optical interconnects offer several well-known advantages such as higher spatial and temporal bandwidths, lower cross-talk independent of data rates, higher interconnect densities, better signal integrity at high frequencies, lower signal attenuation, and lower power requirements at higher bit rates,[3,4] all of which could potentially improve the performance and reduce the power dissipation of HPC systems.

Modeling and simulation play a pivotal role in any HPC system's design.[5] CAD tools are essential to optimize design and system parameters to reduce the fabrication cycle time and end-product cost. Although electronic system simulation has made significant progress, the optoelectronics modeling field has lagged behind due to a lack of simulation methodology and tools. Although optoelectronic tools exist that can help us simulate and design optical interconnects, they are either suitable for optical-link-level but not for system-level simulation, or they are intended for electrical interconnects and are used for lack of tools more suitable for optical interconnects' unique needs.

We need to address the end-to-end system design and simulation of optical interconnects for HPC systems for intra-board, board-to-board, and backplane applications at different levels of abstraction—namely, the functional or link and system levels shown in Figure 1. Prior research work in the optoelectronics simulation field has primarily focused either on the link or

Published by the IEEE Computer Society

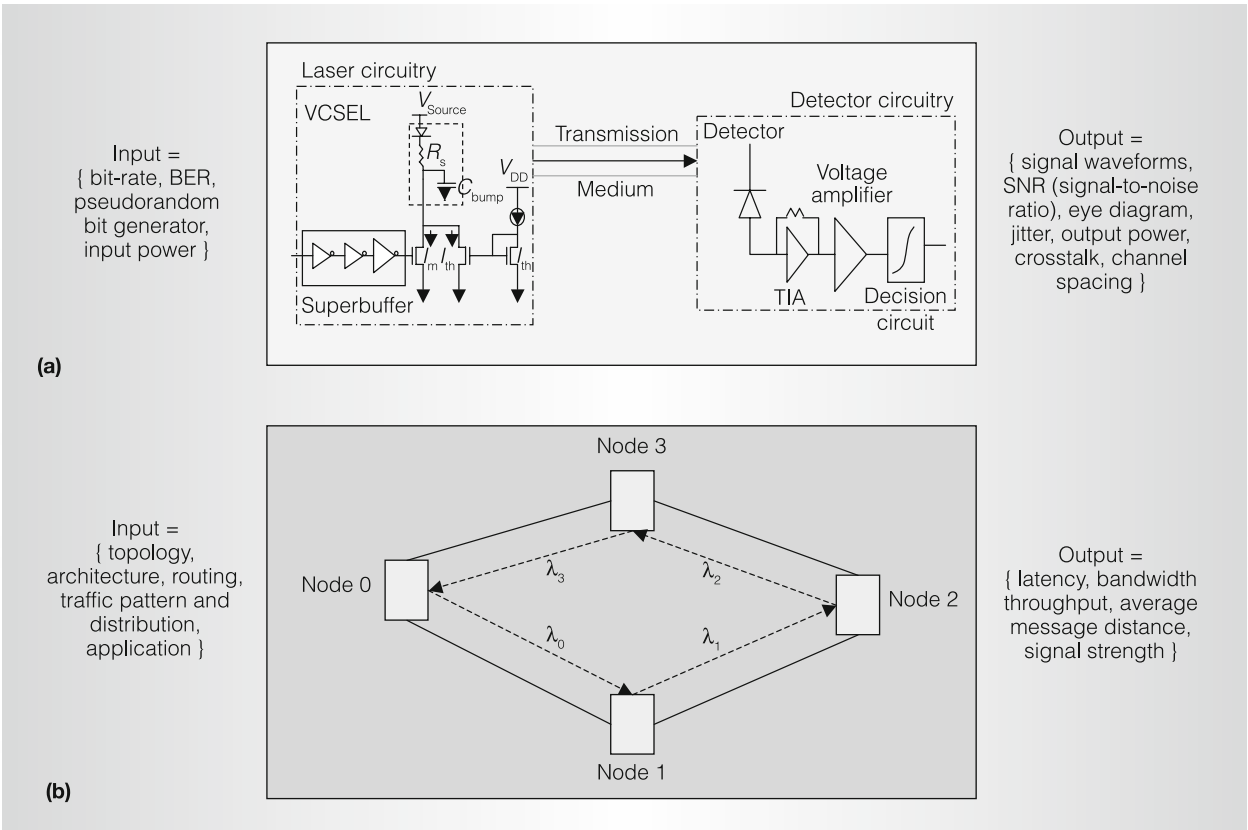


Figure 1. Optical interconnect simulation methodology at functional and link level (a) and the system level (b).

functional levels. The necessary results (or outputs shown in Figure 1a) from an optical link's simulation include signal waveforms, eye diagrams, deterministic and random jitter, signal-to-noise ratios (SNRs), crosstalk, and other output parameters. To obtain these results, we must perform waveform simulation at a given bit-error rate (BER), bit rate (frequency), and input signal power.

Several simulation tools for link-level modeling provide the flexibility to model complex optoelectronic links from the laser to the photodetector taking into account mechanical, electrical, and thermal interactions. (For example, see iFrost,[6] OETC,[7] Chatoyant,[8] and SHAMAN.[9]). However, from the perspective of HPC systems, they do not provide quantitative metrics regarding the system-level optoelectronics simulation parameters such as latency, bandwidth, throughput, average message distance, and signal strength, as shown in Figure 1b. For example, the architecture, topology, routing and wavelength allocation (RWA), and traffic distribution can significantly affect system parameters such as the average network latency, the offered throughput, and power loss. Moreover, the link-level simulation methodology is incompatible with computer architecture simulation. Additionally, optical system design tools such as OPNET and OptiSystem are primarily geared toward telecommunication applications and cannot simply be used for HPC applications. Even though the optical link simulators are useful for functional and link modeling of the optical interconnects, they have limited capabilities in simulating both system-level models for optical devices and optical architectures and topologies.

Optical system designers have evaluated the performance of optoelectronic architectures for HPC systems.[10–12] However, these groups have focused on the optoelectronic architecture and performance trade-offs,

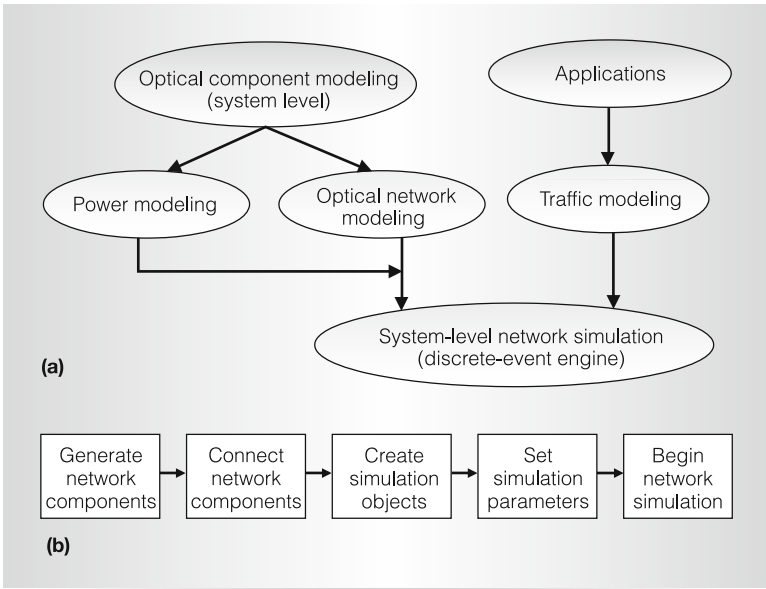


Figure 2. Optisim methodology (a) and flow chart of optical simulation (b).

and there has been no documented simulation methodology available for system designers. To the best of our knowledge, no research to date has presented a detailed simulation methodology that can capture both electrical and optical technology and architecture into an integrated system simulator.

With this article, we propose Optisim, a discrete-event, system-level modeling and simulation methodology of optical interconnects for HPC systems. We have augmented an existing electrical discrete-event simulator by extending its network component library and developed an optical packet simulation methodology. To validate the proposed simulation methodology, we explain it in detail with a case study.

Simulation methodology

In Optisim, we model the optical components and networks at a level of abstraction more suitable for system- than link-level simulation. Optisim is also responsive to traffic patterns, routing, and network architecture. Given that power consumption in interconnection networks is increasing, Optisim models different transmitter and receiver designs, thereby providing power models that can be incorporated at the system level. Optisim has several significant advantages:

- *Efficient component modeling*. We model each optical component or device independently at a level of abstraction that minimizes the computational requirements, while attaining the required system-level simulation accuracy and precision.
- *Accurate latency modeling*. Transmission, propagation, and receiver delays are accumulated to provide accurate optical packet latency.
- *Optoelectronic modeling*. Because future HPC systems will consist of optical components (transmitter, receiver, and medium) and electronic components (buffers, switches, and queues), our proposed methodology incorporates both technologies in the network design to understand performance and cost trade-offs.
- *Optoelectronic power modeling*. Optisim performs power modeling of optical interconnects to evaluate the power consumption for links using different transmitters and receiver designs for varying bit rates.
- *Expandability*. We can easily add active and passive optical components to the simulator based on the number of inputs, outputs, and expected functionality.
- *Extensibility*. The designed optical interconnect simulation framework can be easily integrated with other complex computer architecture system simulators for distributed and parallel computers.

Figure 2a shows our proposed conceptual modeling and simulation framework for optical interconnects.

Optisim models parameterized optical passive and active components and devices as a black box with a set of input and output functions. These modular optical components are recalled from the network library to design the user-specified network topology. To this network model, we add the link's optical power-consumption models. We then extract the traffic model from either an HPC benchmark or a synthetic traffic distribution. Both the modeled network topology and the traffic pattern

are embedded into a system simulation engine. This discrete-event simulation engine can run independently or as a part of a complete computer architecture simulator.

We chose the YACSIM/NETSIM discrete-event simulator, developed by Rice University.[13] YACSIM provides several simulation objects such as processes, events, semaphores, queues, and barriers—basic utilities required for any discrete-event simulator. NETSIM is an electrical network component and simulation library. Combined, YACSIM and NETSIM help us construct a range of direct and indirect electrical interconnects. Using YACSIM as the simulator engine, we augmented the NETSIM library with optical components and optical simulation.

Optical components and architecture design

Figure 2b shows that the first step in designing a system-level optical-interconnect-based simulator is to generate network components. NETSIM includes a library of several electrical components, including ports (packets transmitting and receiving units), buffers (packet storage units), electronic routing units, and electronic switching units. We augmented the NETSIM library with several active and passive optical components such as lasers, couplers, splitters, switches, wavelength converters, waveguides, fibers, multiplexers, demultiplexers, and photodetectors. For system-level modeling, we extract four relevant parameters from the link/functional modeling of each of these components:

- *length*, to determine the propagation latency through the component;
- *attenuation*, to determine the signal loss caused by the component;
- *wavelength*, to determine the routing within a component; and
- *power*, to determine the power the component consumes.

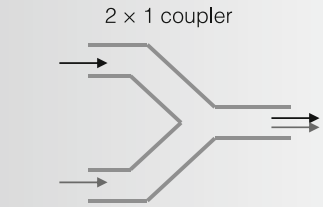
Each optical component is designed with a set of input parameters: **`OpticalComponent < fanin, fanout, length, attenuation, wavelengths, power >`**. **`fanin`** provides the number of inputs to the component, **`fanout`** provides the number of outputs from the component, **`length`** specifies the length in meters, **`attenuation`** refers to the signal loss in dB due to the component, **`wavelengths`** specifies the number of channels the component can transmit, and **`power`** calculates the power the component consumes. In certain optical components such as wavelength converters, output wavelength will be a function of the received input wavelength. The power consumed is calculated based on the type of optical component specified. This value is added only for active optical components such as transmitters, receivers, and other electro-optic devices.

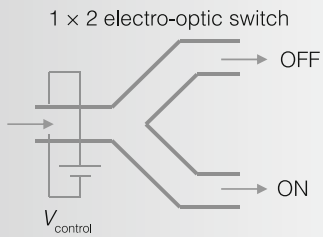
In Optisim, we abstract optical components by capturing key attributes needed for system-level modeling. For example, a transmitter is a single-output device, emitting at a given wavelength λ_k with a certain coupling loss between the laser and the coupling device. Therefore, a laser can be designed as $\text{Optical}_{\text{Transmitter}}(0,\ 1,\ 0.0m,\ 1.0\ \text{dB},\ k,\ p)$, where p is the power the laser and driver circuitry consume. For example, Figure 3a shows a coupler, an electro-optic switch, and a demultiplexer, and Figure 3b shows the sample code snippets.

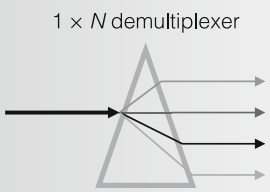
Consider an $N \times 1$ coupler that can couple multiple wavelengths from different inputs N to the single output. A coupler is a passive device and therefore can transmit most of the wavelengths originating from its inputs. It has a largely fixed attenuation, is approximately dependent on the number of inputs ($3 \times \log(N)$ dB), and is approximately 2 to 5 mm long. Therefore, we can characterize the coupler as $\text{Optical}_{\text{Coupler}}(n,\ 1,\ 0.002m,\ 3 \times \log(N)\ \text{dB},\ k,\ 0)$.

Similarly, a $1 \times N$ splitter has the opposite effect, where the same signal is split into n outputs and can be characterized as $\text{Optical}_{\text{Splitter}}(1,\ n,\ 0.002m,\ 3\log(n)\ \text{dB},\ k,\ 0)$. Moreover, we can extend these splitters to design optical switches using some additional device parameter (voltage, temperature, and current). Figure 3 shows a 1×2 electro-optic switch in which the switching is performed based on the applied voltage, V_{control}.

We can extend this simple 1×2 switch to form large, complex switch designs. A demultiplexer acts as a $1 \times N$ switching







(a)

```
OCOUPLER *NewOCoupler(id,fanin,length) {
       OCOUPLER *ocoupler;
       int i;
       ocoupler = (OCOUPLER*)malloc(sizeof(OCOUPLER));
       ocoupler->id = id;
       ocoupler->type = OCOUPLERTYPE;
       ocoupler->fan_in = fanin;
       ocoupler->nextmodule = NULL;
       ocoupler->channel = (int*)malloc(WAVELENGTHS*sizeof(int));
       ocoupler->length = length*(log((double)fanin)/log(2));
       ocoupler->attenuation = 3.0*(log((double)fanin)/log(2));
       return ocoupler;
}
```

```
OSWITCH *NewOSwitch(id, OFF, fanout, length, power) {
       OSWITCH *osswitch;
       oswitch = (OSWITCH*)malloc(sizeof(OSWITCH));
       oswitch->id = id;
       oswitch->type = OSWITCHTYPE;
       oswitch->nextmodule = (MODULE**)malloc(fanout*sizeof(MODULE*));
       oswitch->index = (int*)malloc(fanout*sizeof(int));
       oswitch->channel = (int*)malloc(WAVELENGTHS*sizeof(int));
       oswitch->fan_out = fanout;
       oswitch->control = OFF;
       oswitch->power = POWER_ON;
       return oswitch;
}
```

```
ODEMUX *NewODemux(id, fanout, length,
                        DEMUX_ATTENUATION, WAVELENGTHS) {
       ODEMUX *odemux;
       int i;
       odemux = (ODEMUX*)malloc(sizeof(ODEMUX));
       odemux->id = id;
       odemux->type = ODEMUXTYPE;
       odemux->nextmodule =(MODULE**)malloc(fanout*sizeof(MODULE*));
       odemux->index = (int*)malloc(fanout*sizeof(int));
       odemux->channel = (int*)malloc(WAVELENGTHS*sizeof(int));
       odemux->attenuation = DEMUX_ATTENUATION;
       odemux->fan_out = fanout;
       odemux->length = length;
       return odemux;
}
```

(b)

Figure 3. Modular approach to designing optical components. Example optical components shown include a coupler, a demultiplexer, and a fiber (a) along with sample simulation code (b).

device that switches based on the transmitted wavelength. We have similarly designed a waveguide, fiber, $N \times N$ arrayed waveguide gratings (AWG), and wavelength converters.

From Figure 2b, the next step is to connect the various network components. We now connect modularly designed network components to each other using the OpticalNetworkConnect (src, dest, src_{index}, $dest_{index}$). Here, src is the originating component that is connected to the dest component. From Figure 3b, the nextmodule function embedded within the component's design is used to form this connection. If we must connect multiple components, then depending on whether the concerned component is the src or dest, we use the src_{index} and $dest_{index}$, respectively.

For example, consider a demultiplexer, which routes the packet based on the optical signal's wavelength. Every output is associated with a particular wavelength. src_{index} indicates the correct next module the demultiplexer's output should be connected to. The third step in Figure 2b is to create simulation objects, and the fourth step is to set the simulation parameters, both of which we accomplish using the YACSIM engine.

Optical packet simulation

Each packet is generated with a unique sequence number in the system. In Optisim, we simulate an optical packet using two events (procedures): the head and tail events. Every time an optical packet is ready to be injected into the network, the

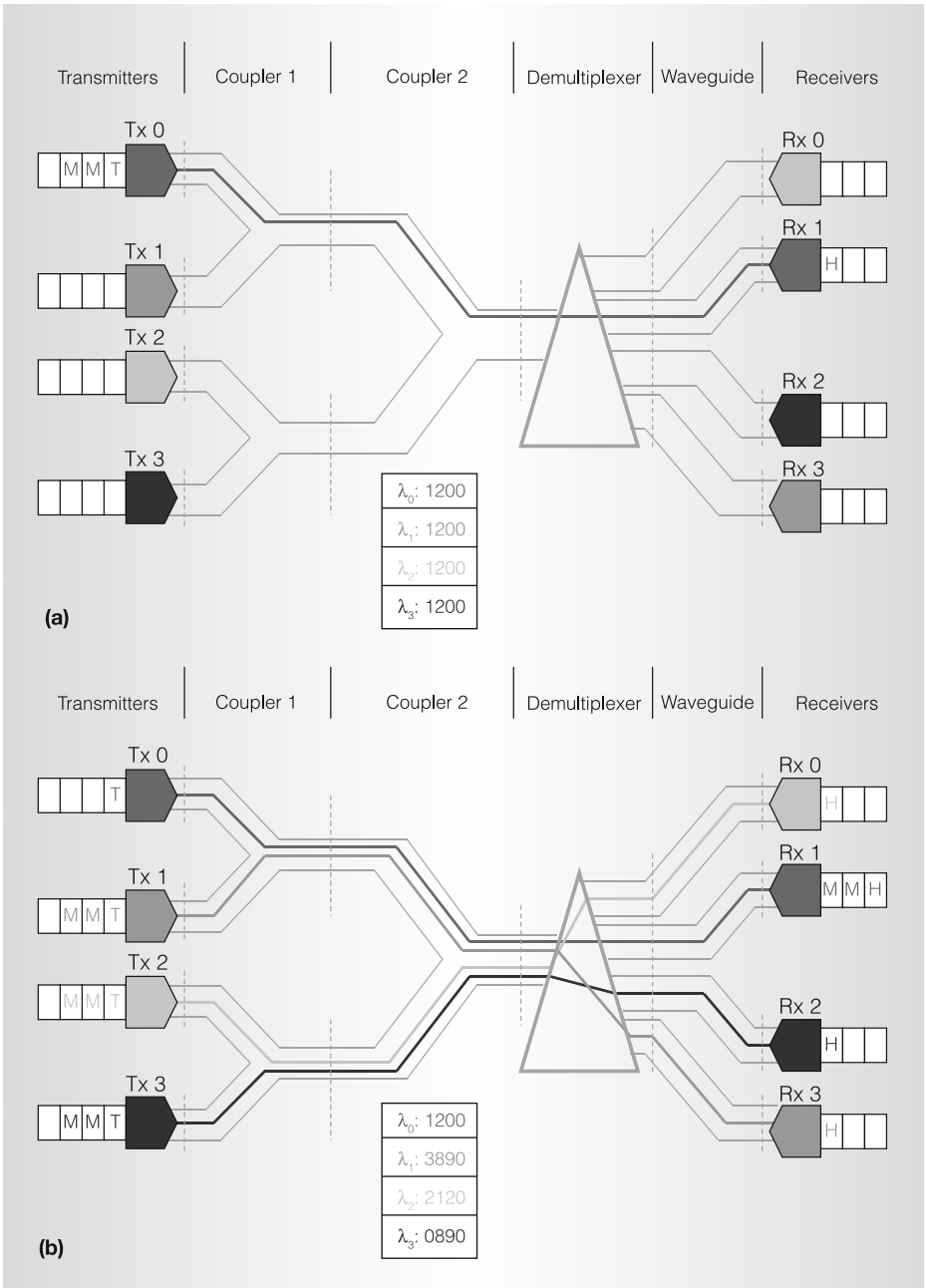


Figure 4. The head flit from Tx 0 reaches the Rx 1 embedding the sequence numbers (1200) within each network component (a). The mid flits from Tx 0 have now reached the receiver, and other transmitters transmit the packet (b).

two events are automatically generated. The optical packet is injected into the network with some attributes, such as the laser's signal strength and the wavelength associated with the transmitter port. The head event immediately sets the path from the source to the destination.

Figure 4 describes the optical packet simulation methodology. This consists of four tunable transmitters and four fixed receivers. Each transmitter is associated with multiple wavelengths (an array of lasers) so that it can reach any of the receivers. Consider the packet transmission from

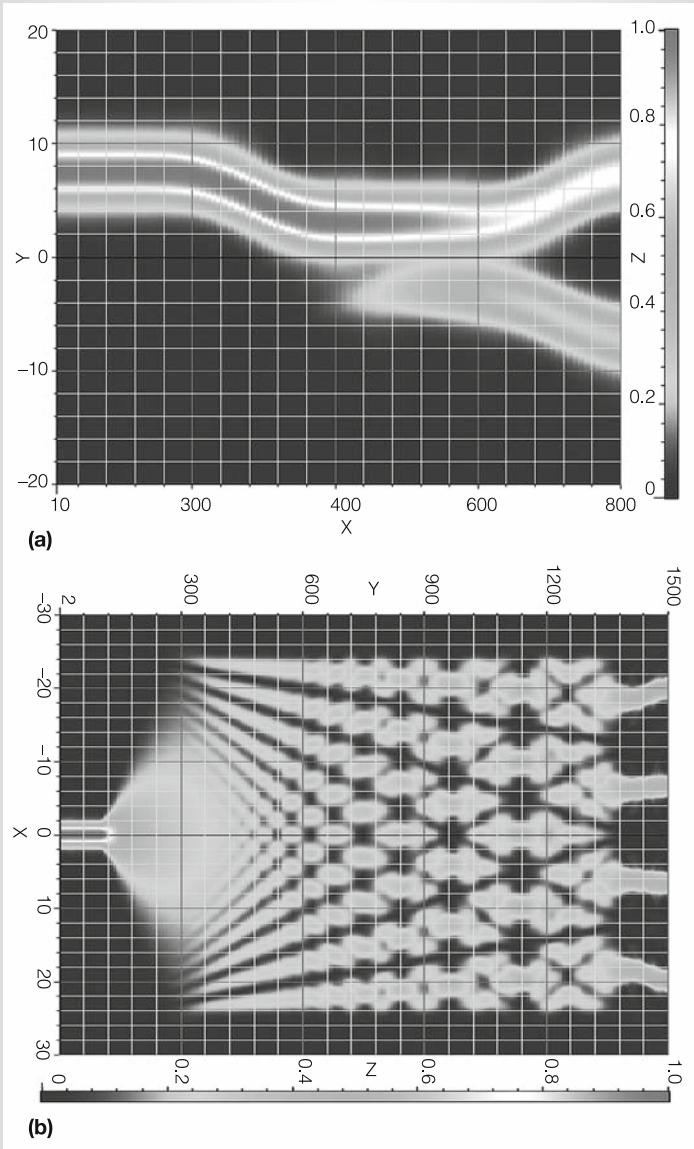


Figure 5. Output from a 3-dB coupler (a) and 1 × 4 splitter (b).

transmitter Tx 0 to receiver Rx 1 on wavelength λ_0, as Figure 4a shows. The head *H* uses the nextmodule function embedded in each component to trace to the next component. The head from Tx 0 traces the route through coupler 1, coupler 2, demultiplexer, waveguide, and receiver. At each component, the head event accrues several component attributes, such as its length, attenuation, and routing. In addition, the packet's head embeds the packet sequence number into the channel the component specifies. When the packet's head from Tx 0 reaches coupler 2, the head event first checks and then embeds the packet's sequence number (1200) into channel λ_0 associated with the component.

Additional features embedded into the component's functionality execute when the head reaches the particular component. For example, consider a splitter that splits the input signal into all its outputs. Here, the head event needs to recreate multiple instances of the packets with similar attributes and restart the simulation for each of the newly generated packets. Once the head event reaches the receiver port, it terminates.

After the tail event *T* is created, it's immediately delayed for the transmission latency and held in the transmitter port. The transmission latency is obtained by dividing the packet size (in bits) with the transmitter's bit rate. Figure 4b shows the packet's mid flits *M* transmitted by Tx 0 having reached the receiver. (A *flit* is the smallest unit of packet transmission, generally consisting of a several bits.) In addition, other head events from transmitters Tx 1, Tx 2, and Tx 3 have reached their respective receivers. The tail event first delays for the propagation latency based on the bit rate and then retraces the same path as the optical packet's head. For each component it traces through the network, the tail event checks whether the packet's sequence number exists. If the sequence number exists at the correct wavelength, then the tail erases the sequence number, thereby tearing down the path. This embedding of the sequence number enhances the proposed model's validity. Moreover, once it reaches the receiver port, it incorporates the receiver processing latency within the simulator.

Power modeling optical interconnects

An optical link's power consumption is becoming as critical as its speed in HPC system design.[11] We provide an analytical framework to capture power consumption that can be incorporated into the system modeling design through power-configura-

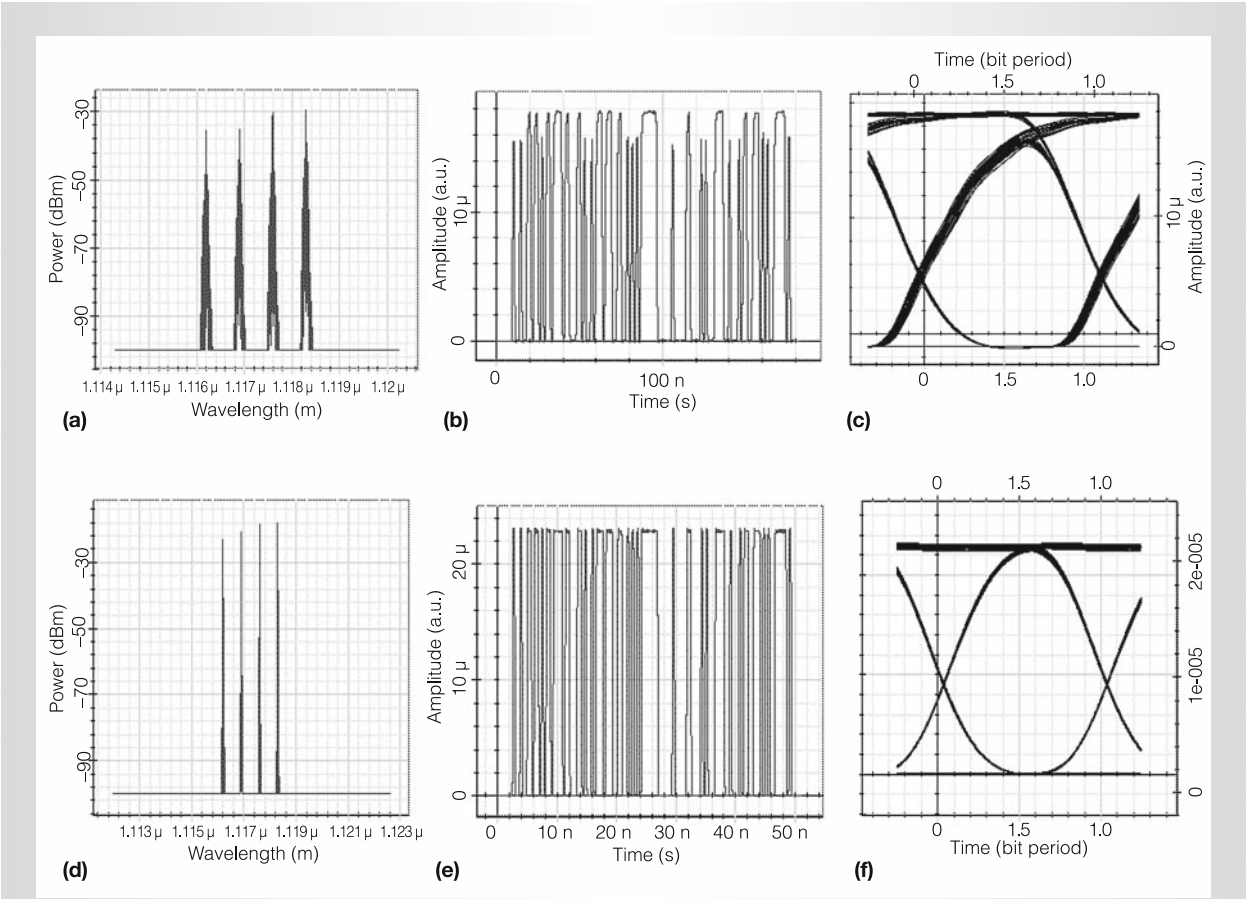


Figure 6. Signal spectrum for a four-channel system at the multiplexer, received data, and eye diagram for 2.5 Gbps using a directly modulated laser source (a–c) and for 10 Gbps using an externally modulated laser source (d–f).

tion files. An optical link consists of the transmitter, receiver, and channel. For a passive channel, an optical link's total power consumption depends on the transmitter and receiver power. Transmitter power is consumed at the laser and its driver and modulator, whereas the receiver power is consumed at the photodetector, transimpedance amplifier (TIA), and clock and data recovery (CDR) circuitry.[11,14]

Multiple-quantum wells (MQWs) with external modulators and vertical-cavity surface emitting lasers (VCSELs) are suitable candidates for laser sources.[14] MQWs need an external laser source to generate light, whereas the light is generated on-chip itself for a VCSEL. For the receiver, we incorporated two designs: low-impedance resistive receiver and TIA-based receiver design. The total power consumed by an entire optoelectronic link is given by

$$\begin{aligned} P_{\mathrm{T}} &= P_{\mathrm{TX}} + P_{\mathrm{RX}} \\ &= (P_{\mathrm{Driver}} + P_{\mathrm{Laser}})_{\mathrm{TX}} \\ &\quad + (P_{\mathrm{Photodiode}} \\ &\quad + P_{\mathrm{TIA}} + P_{\mathrm{CDR}})_{\mathrm{RX}} \end{aligned} \tag{1}$$

We can calculate the total power dissipated in the driver stages as

$$P_{\mathrm{Driver}} = \gamma C_L V_{\mathrm{DD}}^2 B_R \tag{2}$$

where γ is the switching factor, C_L is the total load capacitance, V_{DD} is the supply voltage, and B_R is the bit rate.

In MQW-based modulators, light is received from the external mode-locked

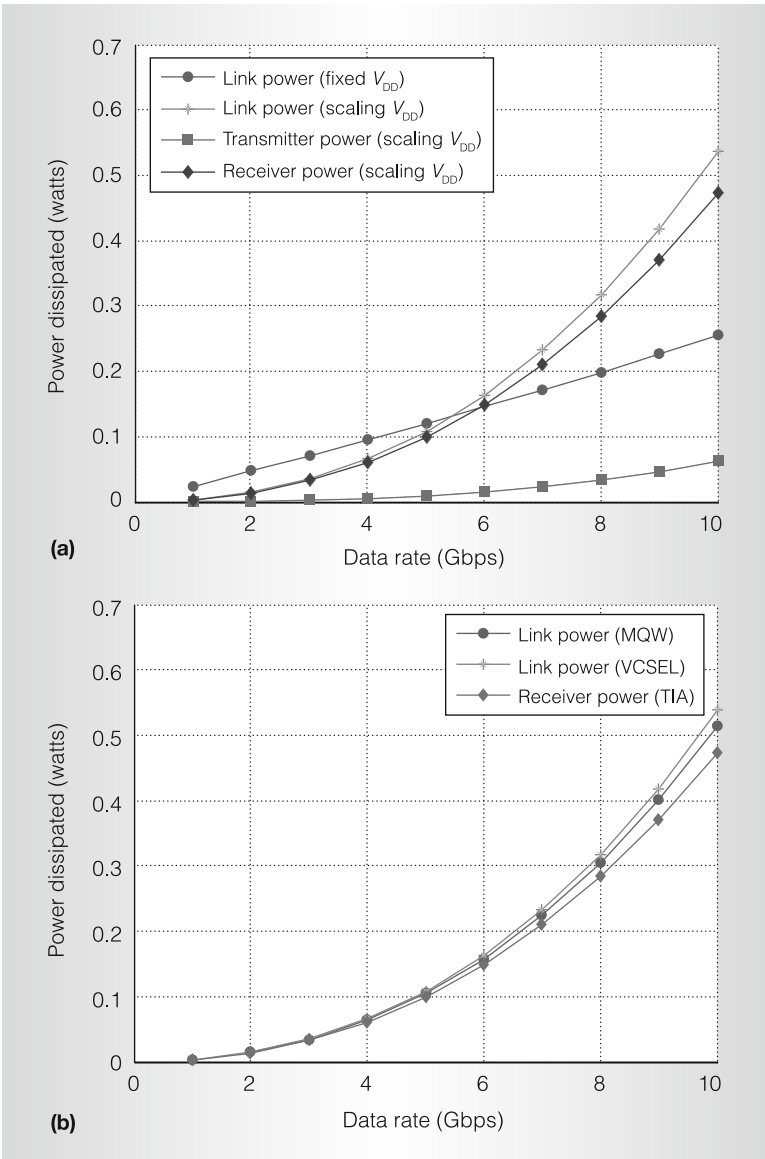


Figure 7. Power consumption for a VCSEL-based configuration with scaling and fixed V_{DD} (a). Power consumption for VCSEL, and MQW-based laser sources (b).

laser. The modulator performance is characterized by its contrast ratio (*CR*), insertion loss (*IL*) at its optimal bias voltage (V_{bias}), and the voltage swing required ΔV_0. The power dissipated in the modulator[15] is given as

$$P_{MQW} = \frac{P_l}{\eta_{link}} \frac{q}{h\upsilon} \left[V_{bias} \left(+ IL - \frac{1 - IL}{CR} \right) - \Delta V_0 IL \right] \quad (3)$$

where P_l is the average optical power required at the receiver input, and η_{link} is the optical system efficiency.

For a VCSEL-based system, we adopt a CMOS driver design,[14] where the driver circuitry consists of two NMOS transistors providing the threshold and modulation currents and a superbuffer driving the gate that delivers the modulation current. The VCSEL power consumed[14] is given as

$$\begin{aligned} P_{VCSEL} &= I_{Total} \cdot V_{source} \\ &= (I_{th} + I_m\gamma)(V_{th} + I_m R_s \\ &\quad + V_{dd} - V_{tn}) \end{aligned} \quad (4)$$

The total current is the sum of the threshold (I_{th}) and modulation currents multiplied by the switching factor. The total voltage is the sum of the VCSEL threshold voltage (V_{th}), the voltage drop across the series resistance (R_s), and the minimum source-drain voltage ($V_{DD} - V_{tn}$) to ensure the gate that delivers the modulation current is in saturation.

For the TIA-based receiver design, we determine the power consumed by the photodetector and the TIA. This consists of the photodetector as a current source ($I_d + \alpha\beta I_m$) and a common source amplifier connected by a feedback resistance, R_f.[16] I_d is the dark current, α is the VCSEL efficiency in ampere/watt (A/W), and β is the detector efficiency in watt/ampere (W/A). The total power dissipated in the TIA based receiver circuit is then given as

$$\begin{aligned} P_{TIA} &= I_b V_{dd} + I_d^2 V_{dd} \\ &\quad + \gamma(\alpha\beta I_m)^2 R_f \end{aligned} \quad (5)$$

where I_b is the internal amplifier's bias current and is given by $I_b = \omega 3_{dBint} V_e C_O$, where $\omega 3_{dBint}$ is the internal amplifier's 3-dB bandwidth, V_e is the early voltage, and C_O is the output capacitance. The power dissipated at the clock and data recovery (CDR) unit[11] is given as

$$P_{CDR} = \gamma C_{CDR} V_{DD}^2 B_R \quad (6)$$

where C_{CDR} is the CDR unit's capacitance. Using this approach, we have modeled two transmitter designs (VCSEL and MQW)

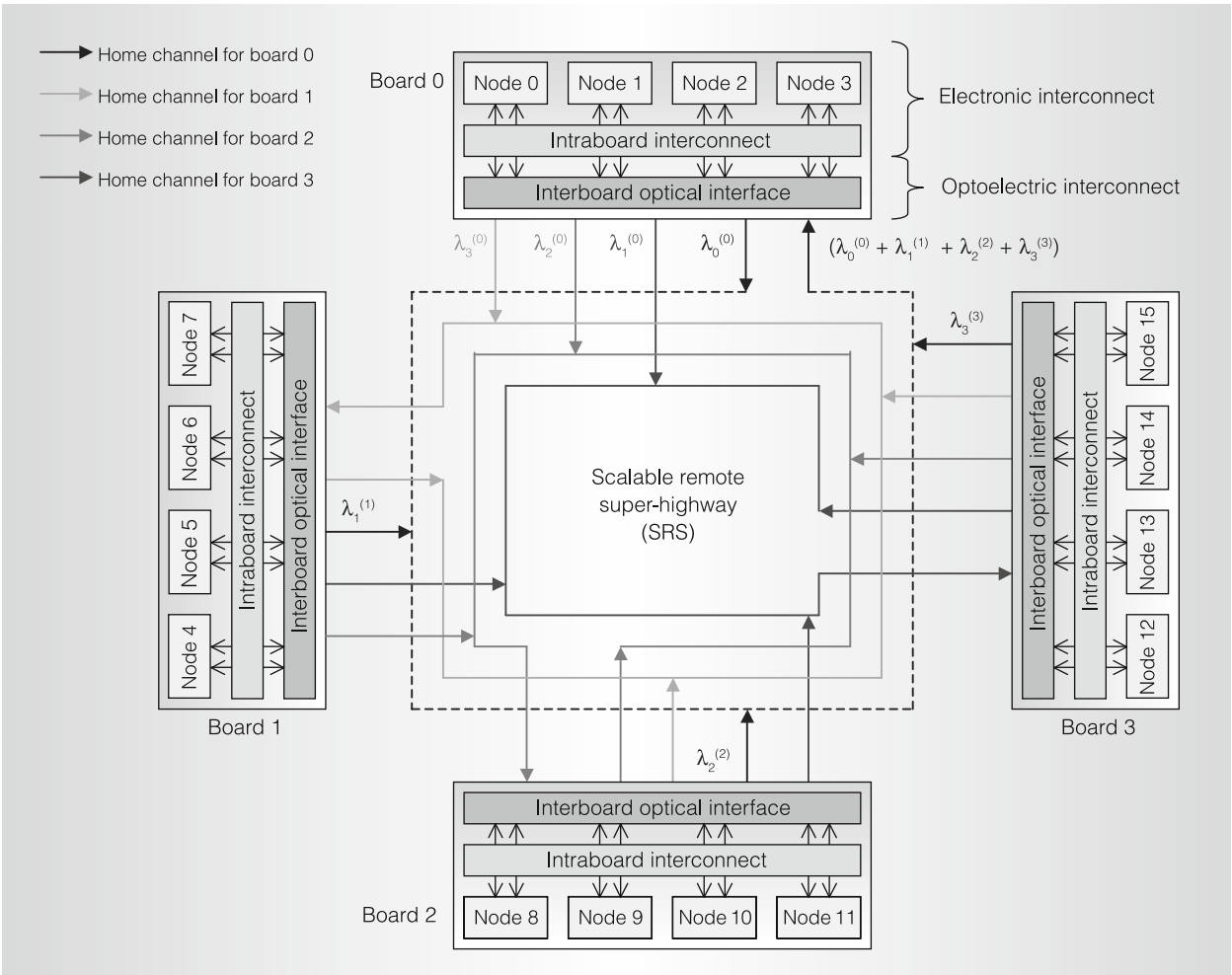


Figure 8. Routing and wavelength assignment for four nodes/boards, four boards, and four wavelength systems.

and two receiver designs (resistive and TIA receivers).

Parameter extraction and system simulation

To evaluate the simulator, we needed to validate the simulation methodology by comparing our approach to a real machine using optical interconnects. However, given the difficulties of testing a real machine and our research's limited scope, we instead extracted parameters from Optiwave's simulation tools OptiBPM and Optisystem (www.optiwave.com) and plugged these into our proposed system simulator.

We designed various optical components and devices using different materials to obtain the desired refractive index contrast, output signal amplitude, and wave propagation using the OptiBPM device-level simulator. We then plugged these discrete components into the OptiSystem link-level simulator to ensure that the eye opening, BER, receiver power, and signal amplitude were sufficient at the specified bit rates and frequency. In addition, we also modeled the power dissipated by the devices and calculated the power consumed by an electro-optic link. This let us test and, to some extent, validate the proposed simulation methodology. Then, we plugged the values (power, attenuation, length, and other parameters) obtained from OptiSystem into Optisim.

OptiBPM. We modeled a 3-dB coupler that was designed using a substrate with

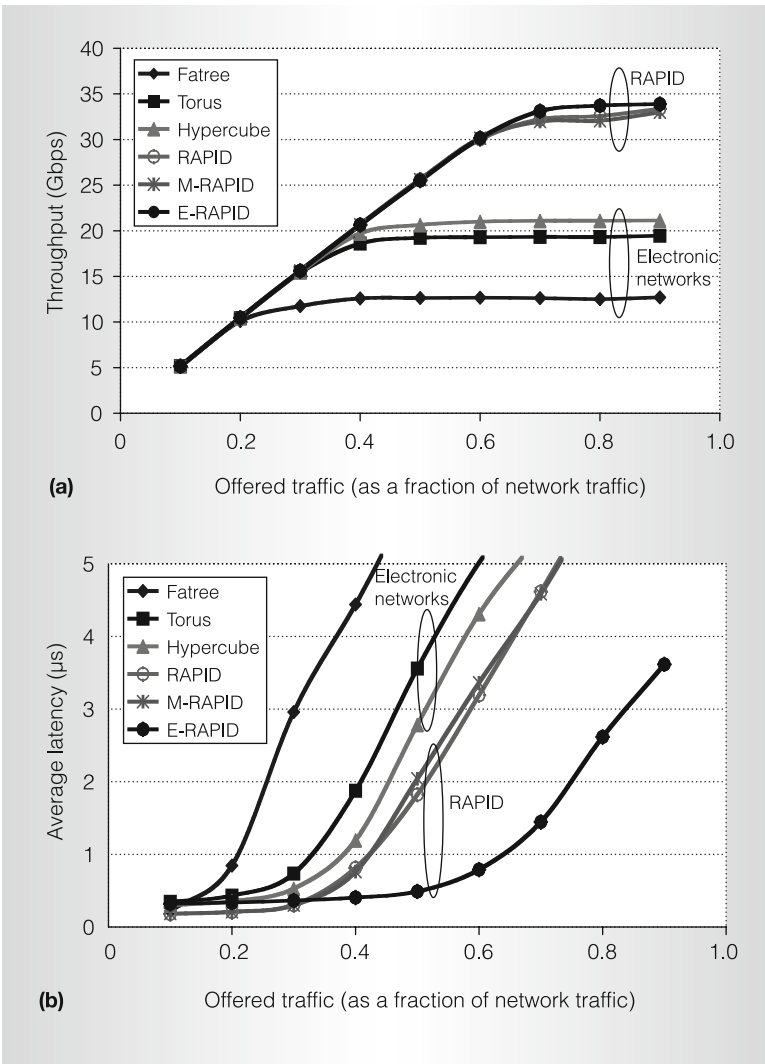


Figure 9. Throughput (a) and latency (b) for uniform communication traffic patterns.

cladding refractive index, n_c of 1.442, and core refractive index, n_r of 1.5. Figure 5a shows this wafer simulation's output. The device was 0.8 mm long, and the output signal had an intensity of 0.45 (approximately 3.3 dB attenuation). We also modeled a 1 × 4 splitter (see Figure 5b). The splitter was 1.4 mm long, and we measured the received intensity at the output at 0.24, implying a 6-dB attenuation. Using the WDM phasar, we evaluated the length of the demultiplexer to be 1.9 mm and an attenuation of 2.1 dB. These parameters were included in the Optisim simulation library's definition files. Currently, we have couplers, splitters, electro-optic switches, wavelength converters, demultiplexers, multiplexers, waveguides, and fibers in our simulator.

OptiSystem. Next, we simulated the OptiBPM parameters using the OptiSystem simulation tool for a four-channel optical-interconnect-based architecture. We solved the directly modulated laser for four channels using rate equations at a data rate of 2.5 gigabits per second (Gbps). The lazing channels were 1116.2, 1116.9, 1117.6, and 1118.3 nm, and the wavelength spacing was 0.7 nm. (*Lazing* implies the wavelength at which the signal is transmitted.) The input power to the laser was 2 mW, or 0.3 dBm. The losses seen by the signal include a propagation loss of 0.2 dB/km, multiplexer loss of 3 dB per stage, and demultiplexer loss of 2.1 dB.

Figures 6a through 6c show the multiplexed spectrum, the received signal, and eye diagram at 2.5 Gbps. The eye diagram shows the eye height of 1.39×10^{-5}, threshold of 9×10^{-7}, and a low BER.

Figures 6d through 6f show the multiplexed spectrum, the received signal, and eye diagram for a four-channel system with continuous wave (CW) laser and an external Mach-Zehneder modulator at 10 Gbps. The eye diagram shows the eye height of 2.21×10^{-5}, threshold of 2.89×10^{-6}, and a low BER. This clearly shows that the four-channel system designed using OptiSystem performs within accepted BER and power budget.

Power estimation. We extracted the parameters for VCSEL and MQW modulators from various sources.[14–16] The link power is dominated by the receiver power consisting of the TIA and CDR, whereas the laser and driver dissipate minimal power. Figure 7a shows the link power for VCSEL-based configuration with fixed V_{DD}, where the supply voltage is not varied; scaling V_{DD}, where the supply voltage is scaled with the bit rates; transmitter power with scaled V_{DD}; and receiver power with scaled V_{DD}. With scaling of bit rates and supply voltages, the power dissipated in a VCSEL is dominated by the receiver

consisting of TIA and CDR. The total power dissipated at 10 Gbps is approximately 535 mW. With the bit-rate scaling from 10 to 5 Gbps and the supply voltage scaling from 1.8 to 0.9 V, the power dissipation for a 5-Gbps link reduces to almost 108 mW.

Figure 7b shows the power dissipated at varying bit rates for a TIA-based receiver for a VCSEL and MQW modulator. MQWs consume marginally less power in Figure 7b because we considered a constant power required at the receiver in the base design.

RAPID case study

As a case study, we considered a reconfigurable all-photonic interconnect for distributed (RAPID) and parallel systems architecture.[17,18] A RAPID network is defined by the (*C*, *B*, *D*) three-tuple, where *C* is the total number of clusters, *B* is the total number of boards per cluster, and *D* is the total number of nodes per board. The total number of nodes in RAPID is the multiplicative factor, $N = C \times D \times B$.

Figure 8 shows the remote wavelength assignment scheme in a *R*(1, 4, 4) system—that is, $C = 1$, $B = 4$, and $D = 4$. For remote communication, different wavelengths from various boards are selectively merged to separate channels to provide high connectivity. Remote wavelengths are indicated by $\lambda_i^{(s,c)}$, where i is the wavelength, s is the source board number, and c is the cluster number from which the wavelength originates. (To clarify, c is dropped because only single cluster working is explained.) The wavelength assigned for a given source board s and destination board d is given by $\lambda^{(s)}{}_{B-(d-s)}$, if $d > s$, and $\lambda^{(s)}{}_{(d-s)}$ if $s > d$, where B is the total number of system boards, the superscript is the source board, and the subscript is the wavelength to be transmitted on. For example, if any node on board 1 needs to communicate with any node in board 2, the wavelength will be $\lambda^{(1)}{}_3$, and for reverse communication, the wavelength will be $\lambda^{(2)}{}_1$.

For example, consider a board 0 transmitter set. All nodes on board 0 have an array of transmitters such that they can transmit on any wavelength $\lambda^{(0)}{}_i$, where $i = 0, 1, 2, 3$. Any node in board 0 can communicate with itself on $\lambda^{(0)}{}_0$, with board 1 on $\lambda^{(0)}{}_3$, with board 2 on $\lambda^{(0)}{}_2$, and with board 3 on $\lambda^{(0)}{}_3$. The physical fiber channel transmits λ_0 on the *home channel* for that particular board (see the dotted line in Figure 8 for board 0). All signals originating from a particular board are demultiplexed and then selectively multiplexed with different home-board channels. For board 0, the multiplexed signal on home channel ($\lambda^{(0)}{}_0 + \lambda^{(0)}{}_1 + \lambda^{(0)}{}_2 + \lambda^{(0)}{}_3$) is then demultiplexed at the board 0 receiver. As the receivers are fixed, λ_i, $i = 1, 2, 3$ are received by node $i - 1$.

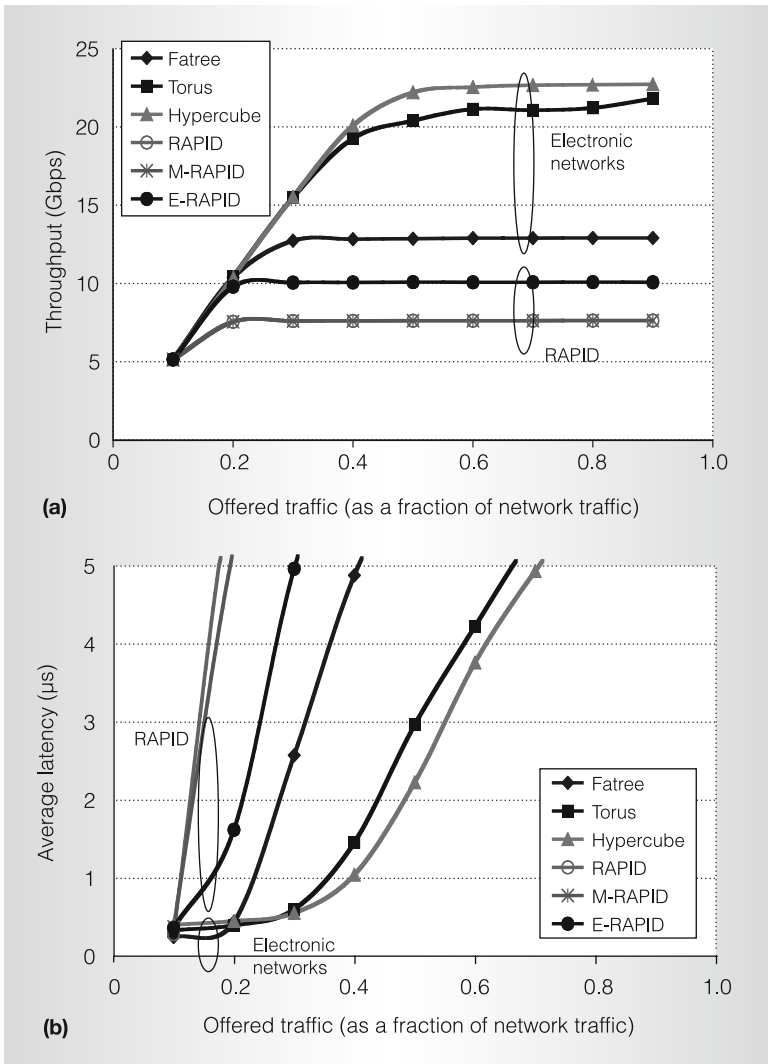


Figure 10. Throughput (a) and latency (b) for the Complement communication traffic patterns.

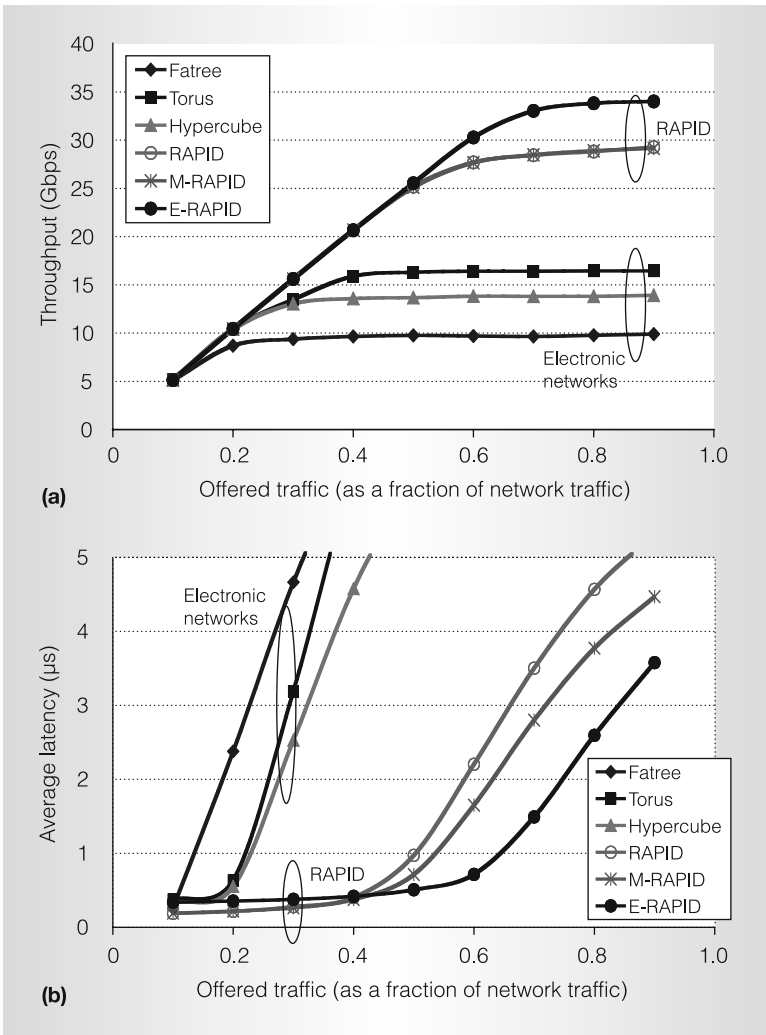


Figure 11. Throughput (a) and latency (b) for the Matrix Transpose communication traffic patterns.

We designed the RAPID network using Optisim simulation methodology as well as multiple transmitters, fibers, demultiplexers, and receivers along with a media access protocol for network simulation. In addition, we designed two cost-effective alternatives, a modified version (M-RAPID) and an extended version (E-RAPID), that minimized the cost of the interconnect based on the number of required transmitters.[18] We then compared the RAPID architectures' performance with various electrical interconnects for uniform, nonuniform, and permutation traffic traces. The electrical networks we chose to compare were 2D torus, hypercube, and fat-tree topologies because they are the most common clustering interconnects. We performed cycle-accurate simulations to evaluate the performance of various topologies for 16 to 1,024 nodes.[18]

Simulation results

We cannot show all the simulation results for various network sizes and traffic traces due space limitations, but Figures 9, 10, and 11 show the throughput and latency for uniform, Complement (node with binary coordinates $a_{n-1}, a_{n-2}, \ldots, a_1, a_0$ communicates with node $a_{n-1}, a_{n-2}, \ldots, a_1, a_0$), and Matrix Transpose (node with binary coordinates $a_{n-1}, a_{n-2}, \ldots, a_1, a_0$ communicates with node $a_{n/2-1}, \ldots, a_0, a_{n-1}, a_{n/2}$) communication traffic patterns, respectively. Figure 9a shows that RAPID configurations outperform all electrical networks, with all RAPID configurations showing almost 30 percent improvement in throughput due to the ample bandwidth provided by optics. With Figure 9b, we can see that the latency for RAPID and M-RAPID saturates at 40 percent of the network load, while E-RAPID shows better performance and saturates at almost 60 percent of the network load.

For complementary traffic patterns, Figure 10 shows that electronic networks outperform RAPID configurations. This is due to the RAPID architecture's design, routing, and wavelength allocation, where all nodes within a board communicate with a particular destination board on a single wavelength. For example, nodes 0, 1, …, 7 on board 0 communicate with nodes 63, 62, …, 56 on board 7 using wavelength $\lambda^{(0)}_1$. This results in high contention, or many accesses at the same time, for the same wavelength by all the nodes within the board, leading to low throughput and high average latency. These results show that system-level modeling and simulation of optical interconnects is crucial to understanding various performance trade-offs. Routing, wavelength allocation, bit rate, signal power, and topology all play a critical role in performance evaluation of system-level optical interconnects.

Figure 11 demonstrates that RAPID configurations show much better perfor-

mance for matrix transpose traffic patterns, with a throughput improvement of almost 100 percent and network saturation of 50 percent, as opposed to 20 percent for electrical interconnects.

In the future, we will integrate Optisim with other fully functional architectural simulators (such as RSIM, SESC, and others) to study various architectural design trade-offs. We will also develop power and thermal modeling of newer optoelectronic components that will benefit the architectural community to analyze the component trade-offs. MICRO

Acknowledgments

This research is supported by US National Science Foundation grants CCR-0538945 and ECCS-0725765.

References

1. E. Mohammed et al., ''Optical Interconnect System Integration for Ultra-Short Reach Applications,'' *Intel Technology J.*, vol. 8, no. 2, May 2004, pp. 115-128.
2. A.F. Benner et al., ''Exploitation of Optical Interconnects in Future Server Architectures,'' *IBM J. Research and Development*, vol. 49, nos. 4/5, Sept. 2005, pp. 755-775.
3. D.A.B. Miller, ''Rationale and Challenges for Optical Interconnects to Electronic Chips,'' *Proc. IEEE*, vol. 88, Jun. 2000, pp. 728-749.
4. J.H. Collet et al., ''Architectural Approaches to the Role of Optics in Monoprocessor and Multiprocessor Machines,'' *Applied Optics*, vol. 39, no. 5, Feb. 2000, pp. 671-682.
5. J.J. Yi and D.J. Lilja, ''Simulation of Computer Architectures: Simulators, Benchmarks, Methodologies, and Recommendations,'' *IEEE Trans. Computers*, vol. 55, no. 3, Mar. 2006, pp. 268-280.
6. B.K. Whitlock et al., ''Simulating Optical Interconnects,'' *IEEE Circuits and Devices*, vol. 11, no. 3, May 1995, pp. 12-18.
7. P.K. Pepelijugoski and D.M. Kuchta, ''Design of Optical Communications Data Links,'' *IBM J. Research and Development*, vol. 47, Mar./May 2003, pp. 223-237.
8. M. Kahrs et al., ''System-Level Modeling and Simulation of 10 g Optoelectronic Interconnect,'' *IEEE/OSA J. Lightwave Technology*, vol. 21, no. 12, Dec. 2003, pp. 3244-3256.
9. Z. Toffano et al., ''Multilevel Behavioral Simulation of VCSEL-Based Optoelectronic Modules,'' *IEEE J. Selected Topics in Quantum Electronics*, vol. 9, no. 3, May/Jun. 2003, pp. 949-960.
10. J.-H. Ha and T.M. Pinkston, ''The Speed Cache Coherence for an Optical Multi-Access Interconnect Architecture,'' *Proc. 2nd Int'l Conf. Massively Parallel Processing Using Optical Interconnections* (MPPOI), IEEE CS Press, 1995, pp. 98-107.
11. X. Chen et al., ''Exploring the Design Space of Power-Aware Optoelectronic Networked Systems,'' *Proc. 11th Int'l Symp. High-Performance Computer Architecture* (HPCA 05), IEEE CS Press, 2005, pp. 120-131.
12. O. Liboiron-Ladouceur, B.A. Small, and K. Bergman, ''Physical Layer Scalability of WDM Optical Packet Interconnection Networks,'' *IEEE/OSA J. Lightwave Technology*, vol. 24, no. 1, Jan. 2006, pp. 262-270.
13. J.R. Jump, *YACSIM Reference Manual*, Dept. of Electrical and Computer Eng., Rice Univ., 1993.
14. O. Kibar et al., ''Power Minimization and Technology Comparisons for Digital Free-Space Optoelectronic Interconnections,'' *IEEE/OSA J. Lightwave Technology*, vol. 17, no. 4, Apr. 1999, pp. 546-555.
15. H. Cho, P. Kapur, and K.C. Saraswat, ''Power Comparison between High-Speed Electrical and Optical Inter-Connects for Interchip Communication,'' *IEEE/OSA J. Lightwave Technology*, vol. 22, no. 9, 2004, pp. 2021-2033.
16. A. Apsel and A.G. Andreou, ''Analysis of Short Distance Optoelectronic Link Architectures,'' *Proc. 2003 Int'l Symp. Circuits and Systems* (ISCAS 03), vol. 4, IEEE Press, 2003, pp. 840-843.
17. A.K. Kodi and A. Louri, ''Design of a High-Speed Optical Interconnect for Scalable Shared Memory Multiprocessors,'' *IEEE Micro*, vol. 25, Jan./Feb. 2005, pp. 41-49.
18. A.K. Kodi and A. Louri, ''Rapid for High-Performance Computing: Architecture and Performance Evaluation,'' *OSA Applied Optics*, vol. 45, no. 25, Sept. 2006, pp. 6326-6334.

Avinash Karanth Kodi is an assistant professor of electrical engineering and computer science at Ohio University. His research interests include computer architecture, optical interconnects, chip multiprocessors, and networks on chip. Kodi received his PhD in electrical and computer engineering from the University of Arizona, Tucson. He is a member of IEEE.

Ahmed Louri is a professor of electrical and computer engineering at the University of Arizona and the director of the High-Performance Computing Architectures and Technologies (HPCAT) Laboratory. His research interests include computer architecture, parallel processing, optical interconnection networks, and networks on chip. Louri received his PhD in computer engineering from the University of Southern California. He is a senior member of IEEE and a member of the Optical Society of America.

Direct questions and comments about this article to Avinash Karanth Kodi, Electrical Engineering and Computer Science, Ohio University, 322D Stocker Center, Athens, OH 45701; kodi@ohio.edu.