Schematic Driven Silicon Photonics Design
(Invited)

Lukas Chrostowski[a], Zeqin Lu[a], Jonas Flueckiger[a], James Pond[b], Jackson Klein[b], Xu Wang[b], Sarah Li[a], Wei Tai [a], En Yao Hsu[a], Chan Kim[a], John Ferguson[c], Chris Cone[c]

[a]Department of Electrical and Computer Engineering,
University of British Columbia, Vancouver, British Columbia, Canada
[b]Lumerical Solutions Inc., Vancouver, British Columbia, Canada
[c]Mentor Graphics Corporation, Portland, Oregon, USA

ABSTRACT

Electronic circuit designers commonly start their design process with a schematic, namely an abstract representation of the physical circuit. In integrated photonics on the other hand, it is very common for the design to begin at the physical component level. In order to build large integrated photonic systems, it is crucial to design using a schematic-driven approach. This includes simulations based on schematics, schematic-driven layout, layout versus schematic verification, and post-layout simulations. This paper describes such a design framework implemented using Mentor Graphics and Lumerical Solutions design tools. In addition, we describe challenges in silicon photonics related to manufacturing, and how these can taken into account in simulations and how these impact circuit performance.

Keywords: silicon photonics design, photonic integrated circuits design, design methodologies, electronic design automation (EDA), manufacturing variability, corner analysis

1. INTRODUCTION

In electronic circuit design, the complexity and scale of designs necessitated multi-user collaboration. Namely, a team of designers with different roles, such as concept and system architect, simulation expert, and layout designer, is required to implement a chip design. In addition, there is another team, associated with the foundry that fabricates the designs, that includes device designers and process design kit developers. The typical electronic design process begins with a schematic, namely an abstract representation of the circuit. This requires that the foundry provides device models and layouts, embedded in a "Process Design Kit" (PDK) that can be used by circuit designers for their circuit simulations and to create their layout. This partitioning of tasks is illustrated in Figure 1.

In integrated photonics design on the other hand, it is often one person who is responsible for all stages of the design. Furthermore, the design process is usually "bottom-up", where the design begins with the physical components, and gradually builds up into a system. In order to build large integrated photonic systems, it is crucial to 1) have a design methodology that allows for teams of designers with various roles to collaborate, and 2) have a capability that includes abstract representations (e.g., schematics) with a link to the physical implementation (layout).

The proposed methodology is based on the electronic IC ecosystem. This includes simulations based on schematics, schematic-driven layout, layout versus schematic verification, and post-layout simulations. We have implemented a design framework using Mentor Graphics and Lumerical Solutions design tools that implements the above requirements, as described Refs. 1, 2.
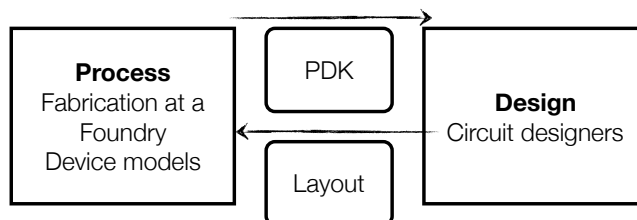

Figure 1: The partitioning of tasks between the foundry and the circuit designer.

This paper expands upon the design methodology described previously by adding:

1. Post-layout simulations

2. A choice in design style – Schematic-Driven Design versus Layout-Driven Design

3. Manufacturing variability simulations (Monte Carlo)

In particular, this paper emphasizes the importance of post-layout simulations. We describe challenges in silicon photonics related to manufacturing, how manufacturing variability can be taken into account in post-layout simulations, and how these simulations can predict the impact of variability on circuit performance.

2. SCHEMATIC-DRIVEN DESIGN

First, we review the schematic-driven design approach described in Refs. 1, 2. This design flow is aimed at highly-complex silicon photonic circuits, analogous to how they are designed in the electronics industry. Here, the design flow starts with an abstract representation, i.e., a circuit schematic. We begin with symbols for the components, place them on a schematic, and connect them. This allows us immediately to simulate the circuit by providing the desired stimulus. Behind the scenes, there are component models. The designer does not need to develop them, since they are provided in the design kit. Advanced PDKs will include active optical components, and electronics.

In this flow, the layout is created on the basis of the schematic. The software aids the designer in creating the layout by reading the schematic as the layout is generated; specifically component placement can be inferred from the schematic, and the connectivity between components is maintained to aid in the routing.

The verification step involves two parts: 1) manufacturing verification (minimum features size, etc), known as Design Rule Check (DRC), and 2) functional verification, known as Layout Versus Schematic (LVS). The latter is critical for large circuits. Since we have a schematic and a layout, we have the opportunity to compare them. The LVS tool uses a set of rules to recognize the components and their connectivity in the layout, known as netlist extraction, and then compares the components and connectivity with the original schematic. This verification step can identify incorrect connectivity, such as disconnected wires and waveguides, and component errors such as incorrect components or incorrect parameters.

Once the layout of the circuit is deemed to be correct, the LVS tool can extract the parameters from the layout and update the schematic. For photonics, the lengths of your waveguides, as drawn, is often critical in determining the performance. For example, the lengths of the waveguides in an interferometer need to be calculated so that we can obtain an accurate estimation and simulation of the free spectral range (FSR). After the schematic is updated, the designer can re-simulate the circuit to verify that it operates as expected once the details (such as waveguide length in photonics, or parasitics in electronics) are included.

In contrast to the traditional photonics design approach, this design flow has a much more well established design kit, and enables rapid design of complex circuits, without having to spend time on modelling individual components. This type of design has been critical to the success of the electronics industry where circuits have thousands to billions of components.

An evolution that is expected for integrated photonics design is that we are moving towards a methodology similar to CMOS design leading to a distinct skill sets. For example, circuit designers will need to know the minimum of information about how a directional coupler works, but will not need to design them (The main knowledge required is that its function is similar to a beam splitter). However, the circuit designer will be highly attuned to the performance criteria of the splitter (e.g., wavelength dependance of the splitting ratio, insertion loss, power splitting balance, sensitivity to manufacturing variations) and how these impact the circuit performance (e.g., extinction ratio, crosstalk). This enables the circuit designer to focus on design at a higher level, including system implementation.

2.1 Post-Layout Simulation

Post-layout simulation refers to circuit simulations that are performed after the layout is created. In the schematic-driven design flow, this involves extracting parameters from the layout, and updating the schematic. However, there is another use case for post-layout simulations. Namely, we desire the capability to simulate directly from a layout without requiring a schematic (e.g., created by a different designer, created using different software, and in general if the schematic is not available).

In this case, the LVS tool can be used to create a netlist representation of the circuit. This is in fact easier to perform than the layout versus schematic operation described above, as it only requires the netlist extraction step. The requirement for this functionality is that each component is supported by the library / PDK. This is necessary so that the LVS tool can recognize the devices correctly, and label them with the appropriate component model. Typically, the components have added marking layers that identify the components, perhaps their names, where the input / output ports are, and provide port names.

After the layout extraction done, the circuit simulator tool is used as before to perform simulations, based on the netlist.

3. LAYOUT-DRIVEN DESIGN

The capabilities developed for post-layout simulation allow us to consider another design methodology, namely layout-driven design. This second approach begins with the layout, as shown in Figure 2. First the designer creates a layout, then simulates it directly from the layout. The simulation is performed by two steps: a) netlist extraction and b) circuit simulations.

This approach may be appealing to traditional photonics designers who have independent design stages: device modelling, circuit design, and layout. In this traditional flow, there is no communication between these stages, hence verification and post-layout simulation is challenging. In the case of separate tools, the need for schematics may not even be obvious or required. For these designers, the benefit of including post-layout simulations in the layout-driven design flow is that accurate simulations can be generated based on the design. An inspection of the simulation results, and comparison with expectations, is an excellent verification step.

Another advantage of the layout-first approach is that it provides insight into what is physically realistic, prior to doing abstract circuit design in a schematic. For example, creating a layout first gives the designer insight into how large the library components are, how long the waveguides will need to be, how far apart the components will need to be, how many waveguide crossings will be required, etc.

Another design option is to create a draft layout with arbitrary component parameters and use the post-layout simulation functionality to create the files necessary for circuit modelling. Once the project is setup with all components and stimulus found in the layout, the designer can proceed to optimize the design within the circuit modelling tool. For example, in a WDM ring modulator transceiver, the choice in ring resonator parameters can be performed in the circuit modelling tool, using the realistic simulation environment created from the layout extraction tool. In such a design flow, it is also important to have the capability for the component parameters to be synchronized between the circuit simulation and the physical layout.

Clearly, there are several design flows that are enabled by the capability of extracting the netlist from the physical layout. The layout-driven design approach is likely to be attractive for small circuit designs.

Both of these design flows (schematic-driven and layout-driven) require that there is a connection between the layout and the schematic, and in both directions. Both need a library of components, containing component models, component layouts, and component layout recognition rules (called an LVS rule deck). The biggest challenge for the designer is to obtain access to all the necessary pieces, namely to have access to a state-of-the art PDK similar to those available in the electronics industry.

4. LAYOUT EXTRACTION

In this section, we provide further details on the layout extraction and netlist generation procedure. As discussed above, extracting the netlist from the layout is important for two reasons: 1) to verify the layout, with respect to the abstract (schematic) representation, 2) to simulate the as-drawn layout, which will include the precise waveguide lengths and
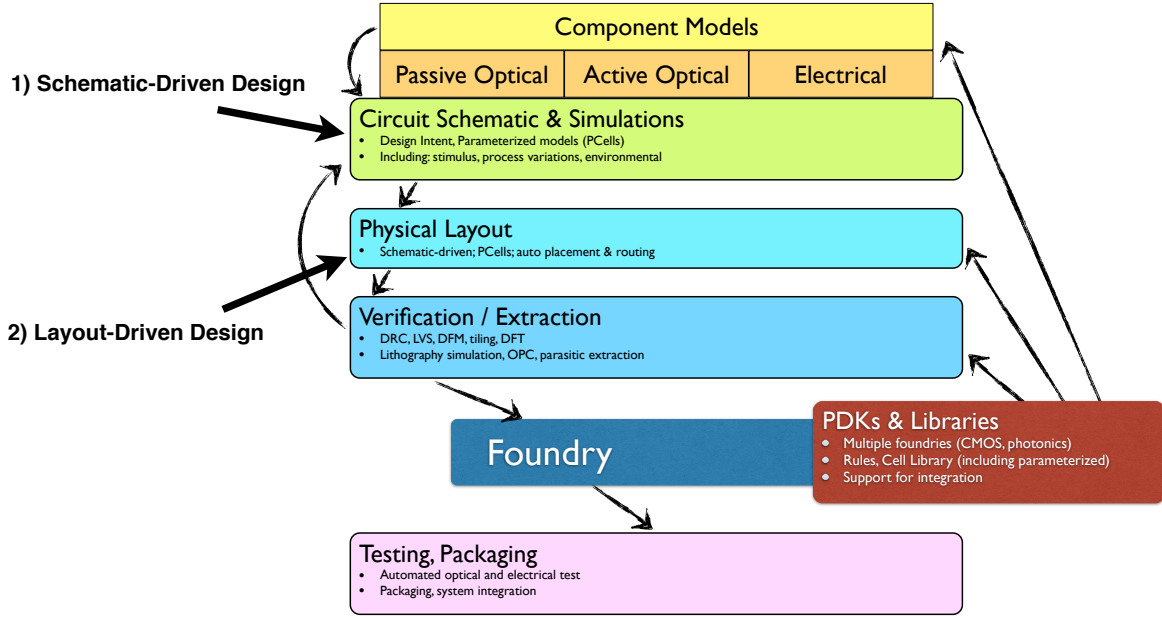
Figure 2: Design methodology based either on 1) Schematic-Driven Design, or 2) Layout-Driven Design.

component position information. The post-layout simulation is an important verification step to ensure that no errors were introduced during the layout (e.g., disconnected waveguides leading to back-reflections).

In this section, we describe an example layout – a Mach-Zehnder Interferometer circuit as shown in Figure 3. It consists of two fibre grating couplers, two broad-band splitters[3], and numerous waveguides and waveguide bends. The netlist is extracted from the layout, either using Mentor Graphics Calibre Layout Versus Schematic (LVS), or using the open-source KLayout[4] tool with the SiEPIC-EBeam-PDK add-on[5]. The extracted netlist is shown in Listing 1. Although this is a simple circuit, the listing is quite long and involves 18 components. Each component includes the following information:

- The nets to which each component pin is connected to (e.g., N$2)

- The library which contains the component's compact model to be used for simulations (e.g., "Design kits/ebeam_v1.2")

- The component position in the layout (lay_x, lay_y)

- Optionally, a position for the component to be used in drawing the schematic (sch_x, sch_y)

- Component parameters, for components that are parameterized. For the waveguides, the parameters are the waveguide width and length. For bends, the parameters are the radius and the waveguide width.

In the above circuit, the waveguide is broken up into straight sections and waveguide bends. The advantage of separately listing these components is that it allows the circuit simulation to be as detailed as possible. For example, the circuit simulation can include:

- An accurate model for the waveguide bend, including the insertion loss, the optical phase through the device, and the reflections from the mode-mismatch from the straight-to-bend region.

- Position information for all these components can be used for thermal modelling and for Monte Carlo simulations, as described below.

Such an approach is particularly useful if the designer has a choice of waveguide bend radius, and the impact of the bends is not known a priori. A simulation with that includes numerous waveguide bends with a too-small bend radius would predict the Fabry-Perot cavities introduced by the small reflections at all interfaces.
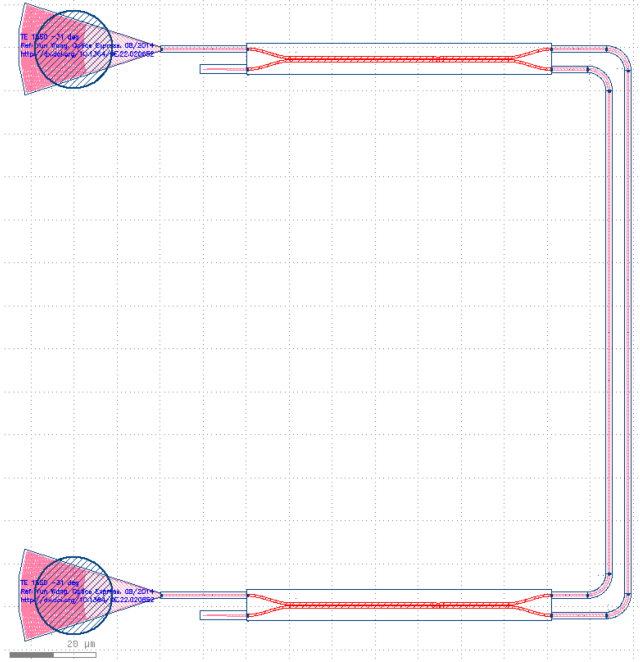
4

Figure 3: Example layout of a Mach-Zehnder Interferometer circuit.

However, in many cases, the bend has already been optimized, analyzed and chosen so that it has little impact on the circuit performance. In that case, precise knowledge of the bend location is not necessary to include in the circuit model, nor is the bend itself. An alternative representation of the above circuit is one where the waveguides and bends are merged into a single waveguide object. This approach reduces the size of the netlist, and improves on the performance of the circuit simulations. One modification is made to the waveguide object as follows:

- The waveguide length includes the straight sections, and the waveguide bends. The length of the waveguide bends is taken as $L = \pi r/2$, where r is the radius taken at the centre of the waveguide.

- The waveguide has an additional parameter, namely a vector of points, e.g., points="((-59050, 5000), (-45690, 5000), (-45690, -117300), (-59050, -117300))" [units of nanometers]. These are used in the Monte Carlo analysis described below.

```
.subckt MZI_circuit N$0 N$1
  ebeam_bdc_te1550_0 N$2 N$3 N$4 N$5 ebeam_bdc_te1550 library="Design kits/ebeam_v1.2" lay_x=-94.35E-6
      lay_y=7.35E-6 sch_x=-4.7175E0 sch_y=367.5E-3
  ebeam_gc_te1550_1 N$0 N$6 ebeam_gc_te1550 library="Design kits/ebeam_v1.2" lay_x=-149.8E-6 lay_y=-117.3E-6
      sch_x=-7.49E0 sch_y=-5.865E0
  ebeam_gc_te1550_2 N$1 N$7 ebeam_gc_te1550 library="Design kits/ebeam_v1.2" lay_x=-149.8E-6 lay_y=9.7E-6
      sch_x=-7.49E0 sch_y=485.0E-3
  ebeam_bdc_te1550_3 N$8 N$9 N$10 N$11 ebeam_bdc_te1550 library="Design kits/ebeam_v1.2" lay_x=-94.35E-6
      lay_y=-119.65E-6 sch_x=-4.7175E0 sch_y=-5.9825E0
  ebeam_terminator_te1550_4 N$3 ebeam_terminator_te1550 library="Design kits/ebeam_v1.2" lay_x=-129.8E-6
      lay_y=5.0E-6 sch_x=-6.49E0 sch_y=250.0E-3
  ebeam_terminator_te1550_5 N$9 ebeam_terminator_te1550 library="Design kits/ebeam_v1.2" lay_x=-129.8E-6
      lay_y=-122.0E-6 sch_x=-6.49E0 sch_y=-6.1E0
  ebeam_bend_1550_6 N$12 N$13 ebeam_bend_1550 library="Design kits/ebeam_v1.2" radius=5.000u wg_width=0.500u
      lay_x=-45.69E-6 lay_y=5.0E-6 sch_x=-2.2845E0 sch_y=250.0E-3 sch_r=180 sch_f=true
  ebeam_bend_1550_7 N$14 N$15 ebeam_bend_1550 library="Design kits/ebeam_v1.2" radius=5.000u wg_width=0.500u
      lay_x=-45.69E-6 lay_y=-117.3E-6 sch_x=-2.2845E0 sch_y=-5.865E0 sch_r=270
  ebeam_wg_integral_1550_8 N$13 N$14 ebeam_wg_integral_1550 library="Design kits/ebeam_v1.2" wg_width=0.500u
      wg_length=112.300u lay_x=-45.69E-6 lay_y=-56.15E-6 sch_x=-2.2845E0 sch_y=-2.8075E0 sch_r=90
  ebeam_wg_integral_1550_9 N$5 N$12 ebeam_wg_integral_1550 library="Design kits/ebeam_v1.2" wg_width=0.500u
      wg_length=8.360u lay_x=-54.87E-6 lay_y=5.0E-6 sch_x=-2.7435E0 sch_y=250.0E-3
  ebeam_wg_integral_1550_10 N$15 N$10 ebeam_wg_integral_1550 library="Design kits/ebeam_v1.2" wg_width=0.500u
      wg_length=8.360u lay_x=-54.87E-6 lay_y=-117.3E-6 sch_x=-2.7435E0 sch_y=-5.865E0 sch_r=180
  ebeam_bend_1550_11 N$16 N$17 ebeam_bend_1550 library="Design kits/ebeam_v1.2" radius=5.000u wg_width=0.500u
      lay_x=-41.34E-6 lay_y=-122.0E-6 sch_x=-2.067E0 sch_y=-6.1E0
```

```
    ebeam_bend_1550_12 N$18 N$19 ebeam_bend_1550 library="Design kits/ebeam_v1.2" radius=5.000u wg_width=0.500u
        lay_x=-41.34E-6 lay_y=9.7E-6 sch_x=-2.067E0 sch_y=485.0E-3 sch_r=270
    ebeam_wg_integral_1550_13 N$17 N$18 ebeam_wg_integral_1550 library="Design kits/ebeam_v1.2" wg_width=0.500u
        wg_length=121.700u lay_x=-41.34E-6 lay_y=-56.15E-6 sch_x=-2.067E0 sch_y=-2.8075E0 sch_r=270
    ebeam_wg_integral_1550_14 N$11 N$16 ebeam_wg_integral_1550 library="Design kits/ebeam_v1.2" wg_width=0.500u
        wg_length=12.710u lay_x=-52.695E-6 lay_y=-122.0E-6 sch_x=-2.63475E0 sch_y=-6.1E0
    ebeam_wg_integral_1550_15 N$19 N$4 ebeam_wg_integral_1550 library="Design kits/ebeam_v1.2" wg_width=0.500u
        wg_length=12.710u lay_x=-52.695E-6 lay_y=9.7E-6 sch_x=-2.63475E0 sch_y=485.0E-3 sch_r=180
    ebeam_wg_integral_1550_16 N$7 N$2 ebeam_wg_integral_1550 library="Design kits/ebeam_v1.2" wg_width=0.500u
        wg_length=20.000u lay_x=-139.8E-6 lay_y=9.7E-6 sch_x=-6.99E0 sch_y=485.0E-3
    ebeam_wg_integral_1550_17 N$6 N$8 ebeam_wg_integral_1550 library="Design kits/ebeam_v1.2" wg_width=0.500u
        wg_length=20.000u lay_x=-139.8E-6 lay_y=-117.3E-6 sch_x=-6.99E0 sch_y=-5.865E0
.ends MZI_circuit

* Optical Network Analyzer:
.ona input_unit=wavelength input_parameter=start_and_stop
  + minimum_loss=50
  + analysis_type=scattering_data
  + multithreading=user_defined number_of_threads=1
  + start=1500.000e-9
  + stop=1600.000e-9
  + number_of_points=2000
  + input(1)=MZI_circuit,N$0
  + output=MZI_circuit,N$1
MZI_circuit N$0 N$1 MZI_circuit sch_x=-1 sch_y=-1
```

Listing 1: Netlist, in SPICE format, for a Mach-Zehnder Interferometer circuit. This netlist includes all bends as separate components.

```
.subckt MZI_circuit N$0 N$1
  ebeam_bdc_te1550_0 N$2 N$3 N$4 N$5 ebeam_bdc_te1550 library="Design kits/ebeam_v1.2" lay_x=-94.35E-6
      lay_y=7.35E-6 sch_x=-4.7175E0 sch_y=367.5E-3
  ebeam_gc_te1550_1 N$0 N$6 ebeam_gc_te1550 library="Design kits/ebeam_v1.2" lay_x=-149.8E-6 lay_y=-117.3E-6
      sch_x=-7.49E0 sch_y=-5.865E0
  ebeam_gc_te1550_2 N$1 N$7 ebeam_gc_te1550 library="Design kits/ebeam_v1.2" lay_x=-149.8E-6 lay_y=9.7E-6
      sch_x=-7.49E0 sch_y=485.0E-3
  ebeam_bdc_te1550_3 N$8 N$9 N$10 N$11 ebeam_bdc_te1550 library="Design kits/ebeam_v1.2" lay_x=-94.35E-6
      lay_y=-119.65E-6 sch_x=-4.7175E0 sch_y=-5.9825E0
  ebeam_terminator_te1550_4 N$3 ebeam_terminator_te1550 library="Design kits/ebeam_v1.2" lay_x=-129.8E-6
      lay_y=5.0E-6 sch_x=-6.49E0 sch_y=250.0E-3
  ebeam_terminator_te1550_5 N$9 ebeam_terminator_te1550 library="Design kits/ebeam_v1.2" lay_x=-129.8E-6
      lay_y=-122.0E-6 sch_x=-6.49E0 sch_y=-6.1E0
  wg0 N$5 N$10 ebeam_wg_integral_1550 library="Design kits/ebeam_v1.2" wg_length=144.727E-6 wg_width=500.0E-9
      sch_x=-2.6185E0 sch_y=-2.8075E0 points="((-59050, 5000), (-45690, 5000), (-45690, -117300), (-59050,
      -117300))"
  wg1 N$11 N$4 ebeam_wg_integral_1550 library="Design kits/ebeam_v1.2" wg_length=162.827E-6 wg_width=500.0E-9
      sch_x=-2.50975E0 sch_y=-2.8075E0 points="((-59050, -122000), (-41340, -122000), (-41340, 9700), (-59050,
      9700))"
  wg2 N$7 N$2 ebeam_wg_integral_1550 library="Design kits/ebeam_v1.2" wg_length=20.0E-6 wg_width=500.0E-9
      sch_x=-6.99E0 sch_y=485.0E-3 points="((-149800, 9700), (-139800, 9700), (-139800, 9700), (-129800, 9700))"
  wg3 N$6 N$8 ebeam_wg_integral_1550 library="Design kits/ebeam_v1.2" wg_length=20.0E-6 wg_width=500.0E-9
      sch_x=-6.99125E0 sch_y=-5.865E0 points="((-149800, -117300), (-139850, -117300), (-139850, -117300),
      (-129800, -117300))"
.ends MZI_circuit

* Optical Network Analyzer:
.ona input_unit=wavelength input_parameter=start_and_stop
  + minimum_loss=50
  + analysis_type=scattering_data
  + multithreading=user_defined number_of_threads=1
  + start=1500.000e-9
  + stop=1600.000e-9
  + number_of_points=2000
  + input(1)=MZI_circuit,N$0
  + output=MZI_circuit,N$1
MZI_circuit N$0 N$1 MZI_circuit sch_x=-1 sch_y=-1
```

Listing 2: Netlist, in SPICE format, for a Mach-Zehnder Interferometer circuit. This is a simplified netlist where the waveguides and bends are grouped into a single waveguide object.
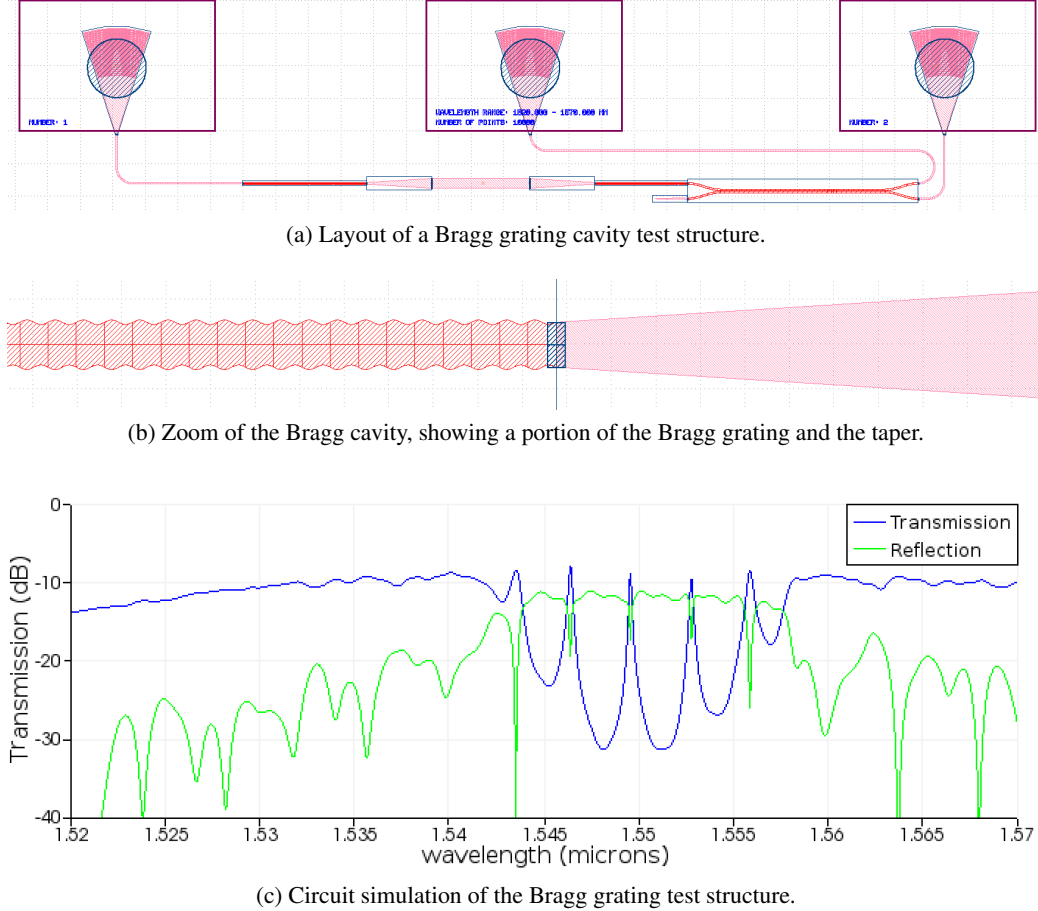
(a) Layout of a Bragg grating cavity test structure.

(b) Zoom of the Bragg cavity, showing a portion of the Bragg grating and the taper.

(c) Circuit simulation of the Bragg grating test structure.

Figure 4: Layout and simulations of a Bragg grating cavity.

5. CIRCUIT SIMULATIONS

We previously described how to perform circuit simulations of photonic circuits using compact models created from S-parameters which are simulated using 3D-FDTD or other numerical techniques[1,2]. This is a powerful technique since it allows any passive device to be represented, without having to establish expressions that describe the behaviour of the component. Efficient circuit simulations can be by achieved by using S-parameters directly, in particular for frequency (wavelength) domain simulations, as the S-parameters are already wavelength dependant. For time domain simulations, the circuit modelling tool internally converts the S-parameters into a time-domain representation.

We provide an example of a Bragg grating cavity test structure layout in Figure 4, together with the simulations from the extracted netlist. The response of the circuit deviates from the ideal Bragg grating cavity response, due: 1) the grating coupler insertion loss, 2) the back-reflections from the grating couplers, and 3) the insertion loss and back-reflections from the tapers. Ripples are observed in both the transmission and reflection spectra, that are not present in the ideal model. Furthermore, the shapes, positions, and oscillation frequencies of the ripples are dependant on the optical phases introduced by the various components, such as the waveguides. Hence it is important to extract the lengths and device parameters for an accurate simulation.

6. MANUFACTURING VARIABILITY

The high refractive index contrast of SOI permits sharp waveguide bends and ultra-small device sizes, making it promising for the development of densely integrated photonic circuits. However, silicon photonic devices face serious manufacturability challenges. Fabrication errors on either the waveguide width or waveguide thickness have a significant effect on the
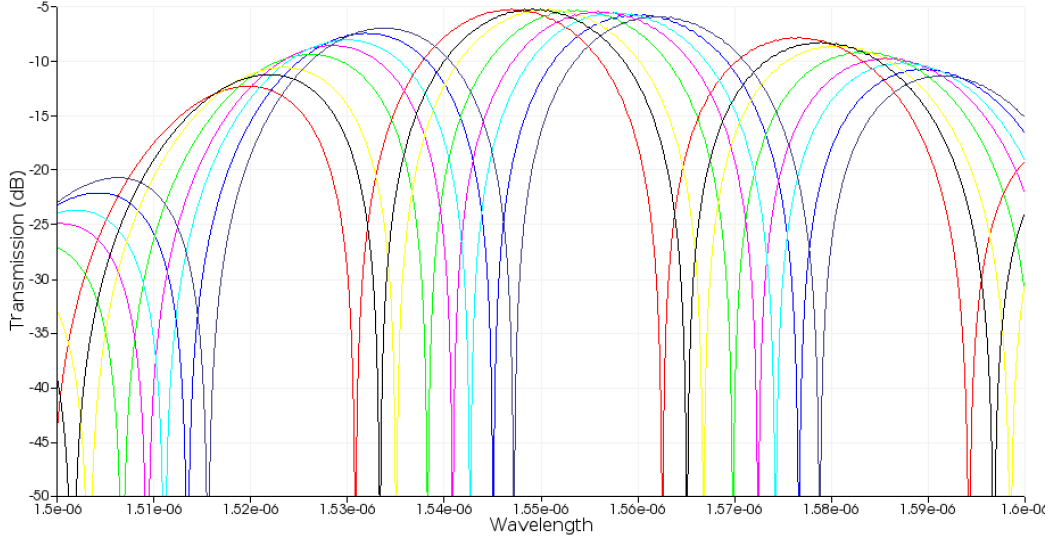
Figure 5: Monte Carlo simulation of the Mach-Zehnder Interferometer layout in Figure 3.

propagation constant of light, and, thus, circuits such as interferometers are highly sensitive to manufacturing, as shown in Figure 5.

Thus, taking manufacturing variability into account in order to predict yield is critical for photonic integrated circuits. In the electronics community, corner analysis and Monte Carlo simulations are typically used to analyze fabrication variability; the former of predicts the worst-case performance of the circuit, while the later creates histograms and distributions, and is based on statistical process variations. In the simplest Monte Carlo approach, the same process variations are applied to all components in the circuit, thus the simulation considers common-mode variability only (e.g., all devices are slightly thicker).

6.1 Component models taking into account variability

In order to perform Monte Carlo simulations, we require parameterized component models that are continuous with the variables of interest (e.g., thickness, width). Although one approach is to simulate each component on a fine mesh (e.g., 0.1 nm steps in each deviation variable), this is computationally prohibitive. We developed more efficient approaches, as follows.

6.1.1 S-Parameter component models

As described in Section 5, we can model components using S-parameters. We begin by generating S-parameters for the corner cases (e.g., components with variations (Δw (nm), Δh (nm)) of (+20, +10), (+20, -10), (-20, +10) and (-20, -10)), as shown in Fig. 6(a). For each Monte Carlo iteration, we interpolate the S-parameters for the desired thickness and width variation. The model can also include multiple modes (e.g., TE and TM polarization).

We use a multidimensional spline interpolation method, which allows for the estimation of the amplitude and phase for each S-parameter taking into account its dependency on multiple parameters, such as the optical wavelength (or frequency), thickness and width. S-parameters passivity and reciprocity can also be tested and enforced, to avoid violating energy conservation laws.

This can be applied to components such as directional couplers, where the phase through the device is dependant on the thickness and width of the waveguide. Two directional couplers can be connected to create a ring resonator sub-circuit, and the performance of the ring resonator (resonance frequency, free spectral range, extinction ratio) will vary with the thickness and width variability introduced into the directional coupler model. As an example, Fig. 6(b) shows the S-parameters for the four process corners, and one example interpolated case for a 2×2 splitter, a device described in Ref. 3.
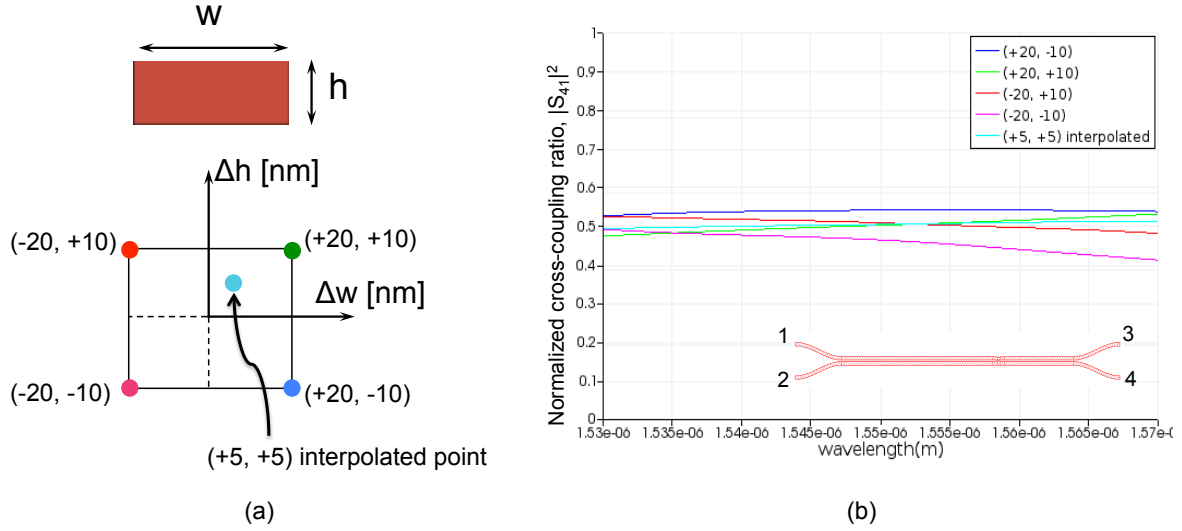
8

Figure 6: (a) Process corners for two process parameters: waveguide height and width; (b) S-parameters for a 2×2 splitter, for the four process corners, and interpolated for an example geometry.
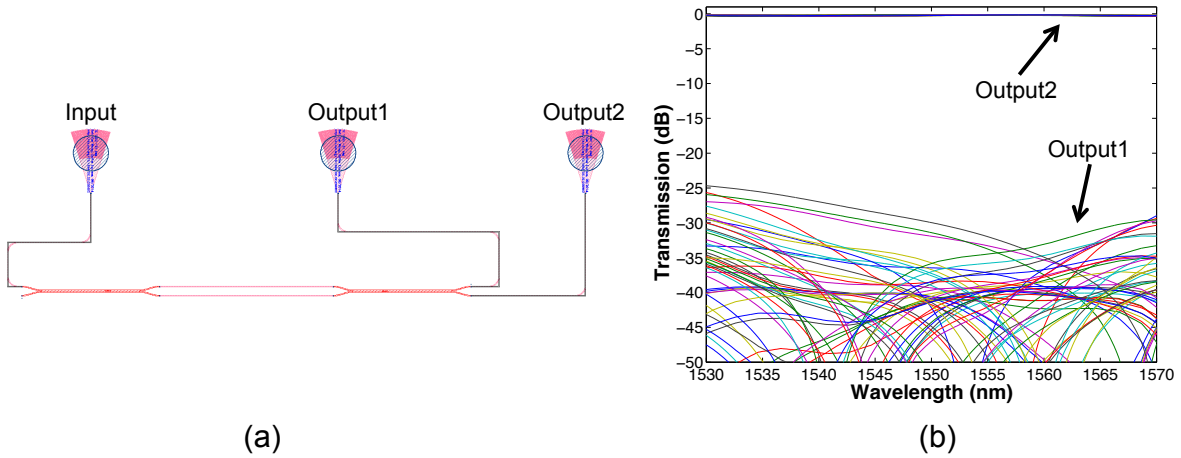

Figure 7: (a) Layout of balanced Mach-Zehnder Interferometer test structure; (b) Monte Carlo simulation results for the output transmissions.

6.1.2 Waveguides

The waveguide transfer function of a waveguide is described by propagation properties such as loss, effective index, group index and dispersion, and need to be parameterized as a function of frequency, thickness and width. We can create parameterized models for the waveguide in a similar manner as in Section 6.1.1, namely using interpolation. We begin with 2D cross-section simulations of the waveguide (eigenmode simulations), and generate look-up tables for the effective index and group index for a range of geometries. We then perform a multidimensional spline interpolation as above.

6.1.3 Sub-circuit approach

For the case of the Bragg grating, for example, we can assume that the manufacturing only affects the central Bragg wavelength. If all other parameters remain the same (grating coupling coefficient) then a small perturbation to the nominal effective index can be applied. This can be expressed as two independent linear parameters, $\frac{\partial n_{\text{eff}}}{\partial w}$, and $\frac{\partial n_{\text{eff}}}{\partial h}$, which are added to the nominal value. Alternatively, we can use the waveguide model above as an input parameter to the Bragg grating compact model. Specifically, the Bragg wavelength, λ_B, is determined by $\lambda_B = 2n_{\text{eff}}\Lambda$, where Λ is the grating period, and n_{eff} is the effective index found in Section 6.1.2. In this latter approach, we can build up sub-circuits using a combination of primitive elements each with their own built-in manufacturing variability.

6.2 Corner Analysis and a simple Monte Carlo

In electronics design, it is common to perform a corner analysis on a circuit. The designer chooses a model from an N-dimensional process variation parameter space, with 2^N choices. Typically, N=2, as a result of two physical variations, and each variable is assigned whether the transistor is Slow (S), Typical (T) or Fast (F). All transistors in the circuit use the same model, namely they all experience the same process variation. While this is useful in electronics in predicting the worst and best case for how fast a chip will operate (e.g., transimpedance amplifier for a detector, or a driver for a modulator), unfortunately this approach is not appropriate for photonics.

First, we demonstrate this approach on a small circuit, Figure 7(a), which is a Mach-Zehnder Interferometer consisting of two 2×2 splitters connected with identical waveguides. We assume that both devices and waveguides experience a common variability. Here, the waveguide width and thickness are varied using a Monte Carlo simulation (where width and height are independent random variables), with each value assigned to each component. The results of the simulations are shown in Figure 7(b), which show that the extinction ratio of the interferometer varies as a result of the manufacturing variability. The results obtained here are independent of the lengths and paths of the two waveguides, provided that the path length difference in the interferometer is $\Delta L = 0$ (i.e., a balanced interferometer). However, we know that from experiments, this is not the case.

Experimental results for balanced interferometers (e.g., travelling wave modulators) show that a "parasitic" free-spectral (FSR) range is observed. In some designs, this FSR can be quite easily observed in the optical spectrum, and corresponds to a large (several π) optical path difference. The same is observed in identical ring resonators fabricated on the same die. The within-die variations lead to a distribution of resonance frequencies. Furthermore, experimental observations show that the variability between devices is correlated to distance between them. For example, two ring resonators closely spaced will have similar resonance frequencies, while those drawn far apart will be highly mismatched. This is common knowledge in analog circuit design, where critical paths are drawn closely together on the mask to minimize the differential error.

In summary, corner analysis and Monte Carlo simulations where the variations are applied in common the the entire circuit are inappropriate for the study of circuit variability, since these do not include the spatial correlations that are experimental observed. They are however useful for predicting individual device variability (e.g., central wavelength of a Bragg grating or ring resonator).

6.3 Monte Carlo with Spatial Correlations

In electronics Monte Carlo simulations, it is possible to simulate circuits where a high degree of correlation exists between the variability of closely placed components. This is included in the model using additional parameters (a mismatch parameter, or a correlation parameter). The challenge for the designer is to determine these parameters based on a physical layout. Ideally, Monte Carlo simulations should capture the die-to-die, device-to-device (intra-die), and intra-device variations, where the within-wafer variations have position-dependant correlations; models for correlation effects have been

developed for electronic circuits[6,7,8,9,10]. For example, for electronic circuit timing analysis, one approach is to group the devices using a partitioning quad-tree approach with each component being assigned several random variables depending on their location in the circuit[6].

In photonics, correlated yield analysis is attracting increasing attention, and there have been several studies investigating the statistical on-chip correlations[11,12], and models for dealing with the uncertainty of devices[13]. From this past work, what is clear is if only common-mode variability is considered (corner analysis, described in Section 6.2 above), the simulations do not capture an important type variability – differential, or spatially correlated variability. Namely, two nominally identical devices side-by-side on a chip will have slightly difference performance. This difference can have significant impact on system performance (e.g., WDM ring resonators, optical switches) and it cannot be simply compensated by thermally tuning the entire chip, but rather requires individual tuning for each component. Clearly tuning every single component in a silicon photonics circuit is prohibitive, hence we require a tool to determine which components need to be tuned. This can be determined by a Monte Carlo analysis that takes spatial correlations into account.

In this work, we propose a methodology that takes correlation between devices into account by using the physical layout in order to perform a manufacturing variability analysis of on-chip photonic integrated circuits. We create a virtual wafer in each Monte Carlo iteration, and each device determines its parameter within the wafer map on the basis of the physical mask layout. Our methodology provides insight into the significance of compact layouts for photonic designs, and offers a useful tool for post-layout verification and power trimming estimation.
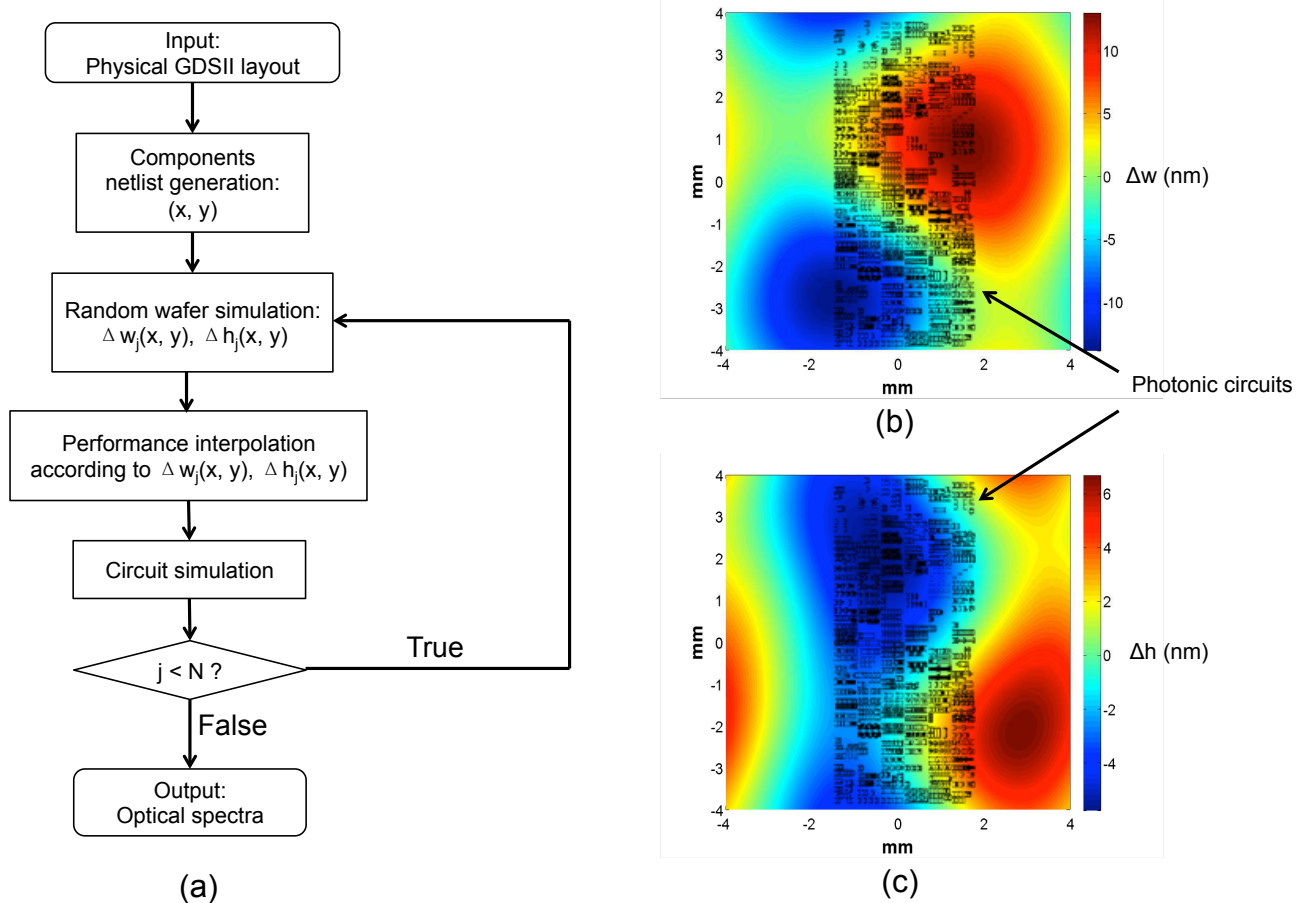

Figure 8: (a) Flow chart of the proposed methodology; (b) simulated waveguide linewidth deviations across a wafer; (c) simulated waveguide thickness deviations across a wafer.

Figure 8(a) shows the flow chart of our proposed methodology. The procedure includes:
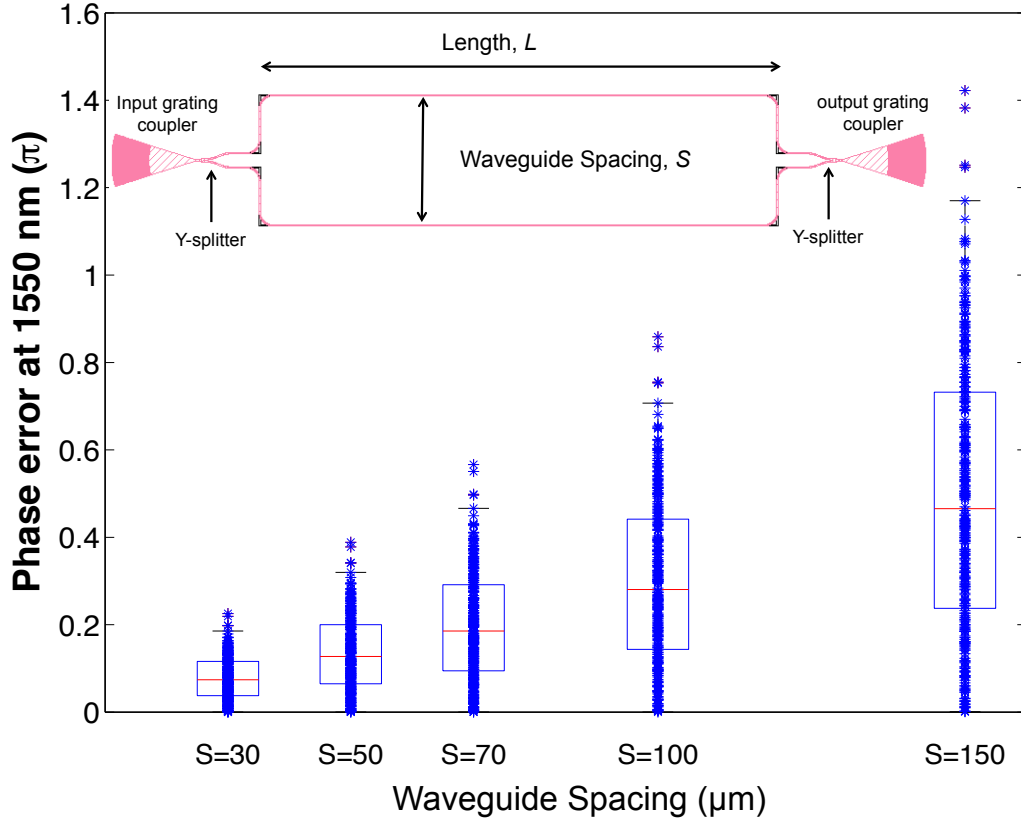
11

Figure 9: Box scatter plot of the phase error of a balanced Mach-Zehnder Interferometer versus waveguide spacing, generated using the Enhanced Monte Carlo simulations.

- First, the layout for silicon photonic circuits and/or devices under test are created. In the example presented here, they are designed using KLayout[4].

- The netlist is extracted from the layout, either using Mentor Graphics LVS or the SiEPIC-EBeam-PDK add-on[5] to KLayout[4]. The netlist provides a list of the primitive components in the library and the connectivity between components. The netlist is imported into Lumerical INTERCONNECT for circuit simulations.

- The third step is the generation of a virtual wafer. Silicon waveguide width deviations, $\Delta w(x, y)$, and wafer thickness deviations, $\Delta h(x, y)$, are simulated across the chip, as illustrated in the maps in Figures 8(b) and 8(c), respectively. Both deviations are characterized by a sigma RMS amplitude and a correlation length, and both are based on experimental results[11,12]. In each run of the Monte Carlo simulation, different deviation maps are be generated in Lumerical INTERCONNECT.

- According to the spatially-dependent waveguide width and thickness map, the optical S-parameters of primitive elements and the optical propagation constants of connecting waveguides are calculated. This is performed using interpolation in INTERCONNECT as described in Section 6.1.1.

- The circuit with components updated based on their positions is simulated. This is repeated several times, and the results can be visualized, as in Figure 5. Histograms and statistical analysis can be performed, as in Figure 9.

The scatter plot in Figure 9 shows the simulated phase error of a balanced Mach-Zehnder Interferometer versus the waveguide spacing between two arms, in a 248 nm silicon photonics process. In the Monte Carlo simulations, the horizontal length of the phase arms, L, is 200 μm. The phase error is defined as the absolute optical phase difference between the

two phase arms. Ideally, the phase error should be 0 when there are no fabrication variations. However, this figure clearly shows that the mean of the phase error for an ensemble of MZIs is approximately linearly proportional to the waveguide spacing, emphasizing the importance of very compact layout designs, particularly where phase matching is required. This simulation is similar to the proportionality observed experimentally in ring resonators[11]. The simulated phase error can be used to estimate the trimming power required for MZI-based photonic switches.

7. CONCLUSION

The objective of this work is "first-time-right" design, where circuits function correctly the first time they are fabricated; this can be achieved only if the designer takes manufacturing challenges into consideration and applies effective mitigation strategies in order to build circuits that are immune to these variations.

ACKNOWLEDGMENTS

References

[1] Chrostowski, L., Flueckiger, J., Lin, C., Hochberg, M., Pond, J., Klein, J., Ferguson, J., and Cone, C., "Design methodologies for silicon photonic integrated circuits," *Proc. SPIE, Smart Photonic and Optoelectronic Integrated Circuits XVI* **8989**, 8989–15 (Feb. 2014).

[2] Chrostowski, L. and Hochberg, M., [*Silicon Photonics Design: From Devices to Systems*], Cambridge University Press (2015).

[3] Lu, Z., Yun, H., Wang, Y., Chen, Z., Zhang, F., Jaeger, N. A. F., and Chrostowski, L., "Broadband silicon photonic directional coupler using asymmetric-waveguide based phase control," *Opt. Express* **23**, 3795–3808 (Feb 2015).

[4] Köfferlein, M., "KLayout," *http://www.klayout.de*.

[5] Chrostowski, L., "SiEPIC EBeam PDK & Library for KLayout," *http://github.com/lukasc-ubc/ SiEPIC_EBeam_PDK*.

[6] Agarwal, A., Blaauw, D., and Zolotov, V., "Statistical timing analysis for intra-die process variations with spatial correlations," in [*Proceedings of the 2003 IEEE/ACM international conference on Computer-aided design*], 900, IEEE Computer Society (2003).

[7] Liu, B., "Spatial correlation extraction via random field simulation and production chip performance regression," in [*Proceedings of the conference on Design, automation and test in Europe*], 527–532, ACM (2008).

[8] Cheng, L., Gupta, P., Spanos, C. J., Qian, K., and He, L., "Physically justifiable die-level modeling of spatial variation in view of systematic across wafer variability," *Computer-Aided Design of Integrated Circuits and Systems, IEEE Transactions on* **30**(3), 388–401 (2011).

[9] Zhang, W., Balakrishnan, K., Li, X., Boning, D. S., Saxena, S., Strojwas, A., and Rutenbar, R., "Efficient spatial pattern analysis for variation decomposition via robust sparse regression," *Computer-Aided Design of Integrated Circuits and Systems, IEEE Transactions on* **32**(7), 1072–1085 (2013).

[10] Liao, C., Tao, J., Zeng, X., Su, Y., Zhou, D., and Li, X., "Efficient spatial variation modeling of nanoscale integrated circuits via hidden markov tree," (2015).

[11] Chrostowski, L., Wang, X., Flueckiger, J., Wu, Y., Wang, Y., and Fard, S. T., "Impact of fabrication non-uniformity on chip-scale silicon photonic integrated circuits," in [*Optical Fiber Communication Conference*], *Optical Fiber Communication Conference*, Th2A.37, Optical Society of America (2014).

[12] Yang, Y., Ma, Y., Guan, H., Liu, Y., Danziger, S., Ocheltree, S., Bergman, K., Baehr-Jones, T., and Hochberg, M., "Phase coherence length in silicon photonic platform," *Opt. Express* **23**, 16890–16902 (Jun 2015).

[13] Weng, T.-W., Zhang, Z., Su, Z., Marzouk, Y., Melloni, A., and Daniel, L., "Uncertainty quantification of silicon photonic devices with correlated and non-gaussian random parameters," *Opt. Express* **23**, 4242–4254 (Feb 2015).