LVS Check for Photonic Integrated Circuits – Curvilinear Feature Extraction and Validation

Ruping Cao*, Julien Billoudet*, John Ferguson*, Lionel Couder*, John Cayo*,
Alexandre Arriordaz*, Ian O'Connor†
*Mentor Graphics Corp.
†Lyon Institute of Nanotechnology, École Centrale de Lyon

Abstract—**This work is motivated by the demand of an electronic design automation (EDA) approach for the emerging ecosystem of the photonic integrated circuit (PIC) technology. A reliable physical verification flow cannot be achieved without the adaption of the traditional EDA tools to the photonic design verification needs. We analyze how layout versus schematic (LVS) checking is performed differently for photonic designs, and propose an LVS flow that addresses the particular need of curvilinear feature validation (curved path length and bend curvature extraction). We show that it is possible to reuse and extend the current LVS tools to perform such critical but non-traditional checks, which ensures a more reliable photonic layout implementation in term of functionality and circuit yield. Going forward, we propose possible future studies that can further improve the flows.**

I. Introduction

Silicon photonics, the art of designing photonic integrated circuits (PICs) on a CMOS platform, has attracted both academic and industrial interest for the possibility of reusing the mature CMOS technology platform that promises high yield and high volume, as well as the potential performance improvement gained by replacing electrons with photons in various application domains. The promise of successful technology integration notwithstanding, another key success factor for the industry is to leverage existing electronic design automation (EDA) tools for photonic designs. However, the issues of adapting an EDA flow developed for electronic IC designs to the specific needs of PIC designs have not yet been fully addressed [1][2].

As part of the EDA flow, a robust and reliable physical verification flow secures the production yield (manufacturability, circuit functionality, reliability, etc.). Physical verification is divided into two main steps: the design rule checking (DRC) and the layout versus schematic check (LVS) comparison. In [3], DRC challenges on non-Manhattan photonic layouts are analyzed, and possible solutions are proposed using an advanced DRC technique.

In this work, we demonstrate the feasibility of LVS for PIC designs where the traditional IC flow cannot be applied directly or perform checks that are specific to photonic circuits. We explain the differences in the LVS verification need between photonic and electronic cricuit, and propose a methodology for performing LVS on PIC that includes curvilinear design validation methods and the associated flow implementations. Finally, we demonstrate with a PIC test case that the developed LVS flow addresses the key missing features in PIC physical verification.

II. LVS on Photonic Designs

After a PIC layout passes through DRC, where physical manufacturability is checked against a set of design rules, it is sent for LVS checking to determine if the circuit behaves as desired – whether a layout implementation of a circuit matches the original schematic design. As part of the LVS job, electrical rule checking (ERC) searches for faulty or dangerous electrical connections. An LVS check on photonic designs is mandatory for the same purpose. However, a dedicated LVS methodology is required due to the different verification requirement.

A. Device Recognition

Electronic layout devices like transistors are extracted based on their layout feature, i.e. an overlap of the polysilicon gate layer and implantation layer can be detected as a transistor instance. In photonics, however, many photonic components are built upon a single waveguide design layer (Fig. 1). In this case, recognition layers and text label are used whenever it lacks an efficient recognition method. The difference also exists in the definition of device. The touching or overlap of layout geometry usually means a continuous signal channel in electrical circuits, but not necessarily in an optical circuit. For example of waveguide crossings (electronically shorted and optically open) and directional couplers (electronically open and optically shorted) [4], those components must be recognized as PIC devices to ensure the signal travels the right path via the defined ports.

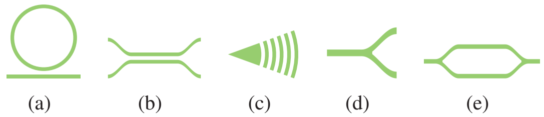


Fig. 1. Curvilinear features of phtotonic designs: layout view of (a) ring resonator, (b) directional coupler, (c) grating coupler, (d) Y-splitter and (e) Mach-Zehnder Interferometer.

Besides the LVS rule deck preparation for device recognition, parameter extraction is not as straightforward due to the curvilinear feature of photonic designs (Fig. 1). While IC device behaviors are characterized by parameters that are measured on Manhattan-like design geometries, such as transistor gate length and width, photonic component design is non-Manhattan, and curvilinear properties such as curvilinear path length and bend curvature are signature features of

978-3-9815370-4-8/DATE15/©2015 EDAA

the components that determine the device function or signal continuity of a waveguide interconnection path [5][6].

a) Path length validation: Manipulation of optical interference behavior enables many important photonic device designs [7][4], like ring resonators, Mach-Zehnder interferometers and arrayed waveguide gratings (AWG). The key parameter to those devices is the path length difference (which corresponds to an optical path length via a certain technology), which must be validated.

b) Bend curvature validation: Another important device parameter is the bend curvature – e.g., a focused grating coupler [8], which is widely used. The curvature of the gratings is a design parameter given by the phase difference between the input wave from the fiber and the output focusing wave. Moreover, the geometrical design of bends determines the signal continuity: radiation loss is directly linked to the bend curvature [9]. Curvature thus needs to be validated to ensure signal integrity, which in this case is in analogous to an ERC check.

Current EDA tools support layout formats like GDSII and OASIS that store layout polygons, which means that curvilinear shapes are rendered into a sequence of straight edges approximating the curve during the discretization process, and the original curve information is lost. Thus the difficulty is in the extraction of curvilinear properties on the polygons, where curvature and path length is not clearly defined. In section III, we show how we perform such extraction and integrate it into a working flow.

III. Method

A. Layout device and connectivity extraction

Using a commercially available tool suite, we define the photonic circuit in the LVS sense as analyzed in section II. A simple circuit is given as an example (Fig. 2), which includes optical components such as grating couplers (GC) for optical signal access from fiber, a Y-junctions (YJ) splitter, ring modulators (RM) for optical signal modulation, bond pads (BP) for electrical signal access, and waveguide (WG) interconnections. Note that the WG interconnection appears as a device, since optical connectivity needs to be verified relying on its geometrical parameters.

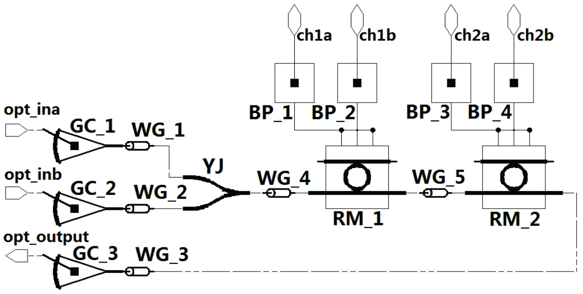


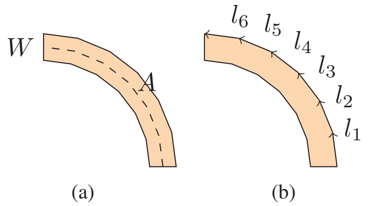
Fig. 2. Schematic design of a photonic integrated circuit.

The implemented layout is extracted and validated for device placement and basic connectivity, which requires a rule deck implementation. The curvilinear property validation, however, requires novel flow implementation.

B. Curvilinear property validation methods

1) Property extraction: A non-disruptive approach to validate the curve designs is to extract properties from the layout curves and compare them with the reference values specified in the source netlist. This flow is similar to the conventional LVS, and the only missing part is the measurement algorithm and its flow integration.

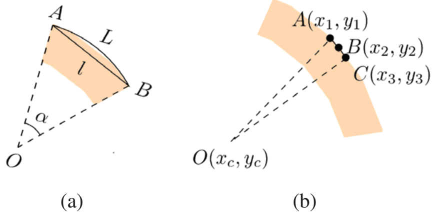
Depending on the design complexity and accuracy requirement, existing syntax can be used that captures the polygon area and width value (Fig. 3(a)). The central path length L of a constant width waveguide routing is computed simply by its area A and width W.



(a) (b)

Fig. 3. (a) Central path length calculation from existing syntax; (b) Contour length calculation from inserted algorithm and modified flow.

It is also possible to estimate the radius of curvature (RoC) with existing tools. On a circular bend, we find the RoC of the curve outer contour R from the the arc length L and the chord length l, which is derived by the captured polygon projection on x axis l_x, on y axis l_y (Fig. 4(a)). Such expressions can all be coded into the rules through built-in languages such as Tcl, and is supported by the existing tools.



(a) (b)

Fig. 4. (a) Curvature calculation on circular bends using existing syntax; (b) curvature calculation on arbitrary bends with a modified flow.

Though path length and curvature can be derived quickly with existing syntax, the limitation is obvious: it works on constant width and circular arcs. Depending on the complexity, these approximations can satisfy certain designs. However, as more complex designs emerge with variant width routing or arbitrary bends, a more advanced approach is required. Here, we use an utility that enables further access to the layout database to let us obtain polygon vertices for measurement.

For length calculation, we sum up the length of linear segments l_i given by the Pythagorean theorem that are applied to each pair of neighboring polygon vertices (Fig. 3(b)), which gives the total side length L. This approach ensures the validity of length extraction on non-constant width or asymmetric designs.

Two general methods exist for RoC extraction from the obtained discrete points: the first method utilizes the discrete data directly, and estimates the required properties; the second method is to bring them back into continuous space by interpolations or parameterizations of mathematical objects, from which the curve properties are computed. Each method can be implemented with a variety of approaches and techniques. For the purpose of demonstrating the flow implementation (the overview and comparison studies can be found in [10] [11] [12]), we applied the osculating circle fitting for the former approach and spline interpolation for the latter. More advanced algorithms can be investigated as well in future study, but scalability and runtime can be a concern, considering the large scale of PIC designs we will have.

For discrete space direct extraction, based on the osculating circle definition of the curvature on a certain point of the curve, we take each group of three points (x_1, y_1), (x_2, y_2) and (x_3, y_3), retrieved in successive manner, from which an osculating circle is defined (Fig. 4(b)). Running the algorithm through all the points on the curve, the local curvature value of each point is obtained. We then output the minimum RoC (that must be controlled against bend loss), and the maximum RoC for comparison with the reference values.

In the case of continuous space extraction, a spline is used for interpolation due to its effectiveness in numerical differentiation[13]. The interpolation results in a continuous curve, with each interpolated segment given by a polynomial function, on which the local curvature and path length are easily computed. The interpolation algorithm is found in [14].

2) Shape comparison: To avoid the extraction difficulty on discrete polygons, an unconventional approach is to reconstruct the curve from the source netlist as a reference design, which is then compared with the layout drawn curve. The curve equations or the coefficients of the parametric curve are required for the reconstruction. To make use of the available SPICE netlist, such information should be coded into a format that is highly restricted. Curves given by equations must explicitly state their form of function and parameters. The generated polygon shape is then written onto the layout as the reference design. We then compare the layout curve with the reference using the geometrical manipulation, i.e., checking if the layout curve contour falls within the reference one.

IV. Results and Discussions

The development is based on sample designs provided in the Generic Silicon Photonics (GSiP) PDK, which is available for download at [15]. Fig 2 and Fig. 5 are the schematic and layout views of a PIC design realized in Pyxis™.

Markers are placed on the waveguide bends (BWG) and the waveguide interconnection path which links two optical component pins (PWG), which may or may not be needed in different flows. The minimum and maximum radius of curvature are defined among other properties in BWG device declaration; and minimum and maximum width, and the path length are defined for PWG.

A. Flow Implementation

1) Flow a: An existing rule-based method is implemented with the left sub-flow (Fig. 6(a)), and it requires no modification to the traditional LVS flow. Computation results of BWG curvature and PWG length as extracted device parameters are validated against values given in the source netlist.

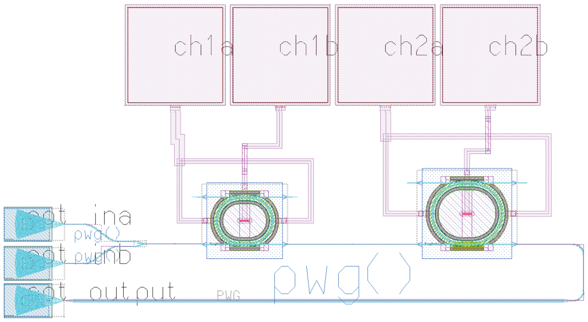
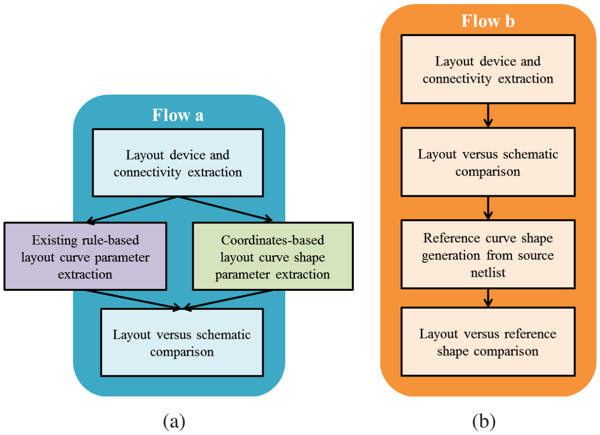


Fig. 5. Layout design of a photonic integrated circuit realized in the Pyxis™design environment.


Flow a
Layout device and connectivity extraction
Existing rule-based layout curve parameter extraction
Coordinates-based layout curve shape parameter extraction
Layout versus schematic comparison
Flow b
Layout device and connectivity extraction
Layout versus schematic comparison
Reference curve shape generation from source netlist
Layout versus reference shape comparison

(a) (b)

Fig. 6. LVS Flows using (a) properties comparison, with two sub-flows for different extraction methods; (b) shape comparison.

The right-hand side sub-flow is implemented for coordinate-based property extraction. The curvilinear parameter extraction task is moved to the ERC stage. It is enabled by the Calibre® PERC™ tool framework[16], which is a reliability verification and analysis platform for physical layout and logical netlist information.

2) Flow b: Shape comparison method is performed with flow b (Fig. 6(b)). We also use the PERC framework to perform source netlist information referencing and manipulation. The reference curve is recovered from the source netlist-specified bend type (circular, sinusoidal), and its corresponding parameters (radius in the circular case, and amplitude, frequency in the sinusoidal case). It is then translated into coordinates and written onto the layout. Finally, the layout drawn curve is compared with the reference curve shape.

B. Comparisons of the Methods

The comparison between the introduced curve validation methods are summarized in table I.

For flow a, the methodology is similar to classical LVS, which compares the property values specified in the source netlist with the layout-extracted ones. Exploitation of existing rules is limited in real-world application, because this technique does not apply to complex PIC designs. Coordinate-

TABLE I
FLOW AND COMPUTATION METHOD COMPARISON.

	Flow a. **Property comparison**		**Flow b.** **Shape comparison**
Flow	Layout curve parameter extraction and comparison with reference values in source netlist		Reference curve generation from source netlist and comparison
Sub-Flow	Existing rule-based extraction	Coordinate-based extraction	N/A
Extraction method	Extraction on layout polygon	Extraction on polygon vertices or reconstructed curve	N/A
Advantage	• No modification or additional tool required	• Extraction on arbitrary curve shapes	• Full compare of arbitrary shape • Extraction difficulty in discrete space avoided
Drawback	• Limited on arbitrary curve cases	• Additional tools required for data acquisition and manipulation	• Extra computation for polygon creation • Unified curve definition method required
Adoption time	• Immediate (rule deck modification only)	• Quick (flow modification and algorithm development)	• Relatively slow (flow modification, standardization)

based extraction provides more flexibility and good accuracy to allow arbitrary curve design, and the flow implementation requires an additional PERC step to enable netlist analysis and data manipulation. Coordinate-based extraction methods can either be performed directly in discrete space, or in continuous space, which avoids the curvilinear feature definition difficulty with the direct extraction method. The coordinate-based method are available with minor development cycle impact, since they do imply minor flow modification and algorithms implementation.

It is interesting to note that downwards in the EDA flow, there exists a process that performs similar continuous contour construction, known as the lithographical contour simulation used in optical process correction (OPC). As a proposal for future study, we can use this information and create a ”Litho versus Schematic” check that can improve validation accuracy while possibly re-using the existing software capability.

For flow b, or the shape comparison flow, curvilinear parameters are not extracted as in flow a. Validation is done by comparison of the layout curve with reference curve geometry. Using the PERC flow, reference curve geometry is recovered from a source netlist (or a complementary file aside to the existing layout format). This also means that we need to restrict the curve description method. Therefore, along with the flow modification and algorithm implementation, a curve information storage format needs be standardized and introduced to the industry. Due to the limitations already found in the SPICE netlist format for handling optical signal information, we require a more capable format that can store curve expressions (e.g.. in the form of splines), for the sake of physical verification. This is suggested as a future area of study.

V. CONCLUSION AND OUTLOOK

This work highlights the demand for an LVS flow that assesses photonic designs. We identified the differences in the LVS flow compared to an traditional IC, and demonstrated how it is adapted for photonic designs using an existing EDA tool suite, especially the unique verification requirement of curved photonic design features. Several methods are proposed to validate layout curve designs, and each of them presents a different level of integration complexity and expected adoption time. Among them, we suggest the coordinate-based interpolation and extraction method as a capable solution – it handles arbitrary curve designs and can be implemented with slight flow modification.

Future work includes the refinement and assessment of the bend curvature extraction algorithms; exploration of further design requirements on the optical signal continuity and verification methodologies; development and standardization of photonic design netlist and design data storage format; and development of a novel LVS flow that validates lithography-simulated contours against the source design.

ACKNOWLEDGMENT

The research leading to these results has received funding from the European Community’s Seventh Framework Programme (FP7/2007-2013) under grant agreement n°318178-PLAT4M; and the French national program ‘Programme d’Investissements d’Avenir, IRT Nanoelec’ ANR-10-AIRT-05.

REFERENCES

[1] W. Bogaerts, “Design Challenges in Large-Scale Silicon Photonics,” in *Desgin Automation Conference*, Austin, 2013, p. 2.

[2] L. Chrostowski, J. Flueckiger, C. Lin, M. Hochberg, J. Pond, J. Klein, J. Ferguson, and C. Cone, “Design methodologies for silicon photonic integrated circuits,” vol. 8989, 2014, pp. 89 890G–89 890G–15.

[3] R. Cao, J. Ferguson, F. Gays, Y. Drissi, A. Arriordaz, and I. O. Connor, “Silicon Photonics Design Rule Checking : Application of a Programmable Modeling Engine for Non-Manhattan Geometry Verification,” in *Very Large Scale Integration (VLSI-SoC), 2014 IFIP/IEEE 22nd International Conference on*, Playa del Carmen, Mexico, 2014.

[4] L. Vivien and L. Pavesi, *Handbook of silicon photonics*. Boca Raton, FL: Taylor & Francis, 2013.

[5] M. Heiblum and J. Harris, “Analysis of curved optical waveguides by conformal transformation,” *IEEE Journal of Quantum Electronics*, no. 2, pp. 75–83, 1975.

[6] Y. Vlasov and S. McNab, “Losses in single-mode silicon-on-insulator strip waveguides and bends.” *Optics express*, vol. 12, no. 8, pp. 1622–31, Apr. 2004.

[7] G. T. Reed, *Silicon photonics: an introduction*. Chichester ; Hoboken, NJ: John Wiley, 2004.

[8] F. Van Laere, W. Bogaerts, D. Taillaert, P. Dumon, D. Van Thourhout, and R. Baets, “Compact Focusing Grating Couplers Between Optical Fibers and Silicon-on-Insulator Photonic Wire Waveguides,” *OFC/NFOEC 2007 - 2007 Conference on Optical Fiber Communication and the National Fiber Optic Engineers Conference*, pp. 1–3, Mar. 2007.

[9] E. Marcatili and S. Miller, “Improved relations describing directional control in electromagnetic wave guidance,” *Bell Syst. Tech. J*, 1969.

[10] M. Worring and A. Smeulders, “Digital curvature estimation,” *CVGIP: Image Understanding*, 1993.

[11] S. Hermann and R. Klette, “A Comparative Study on 2D Curvature Estimators,” *2007 International Conference on Computing: Theory and Applications (ICCTA’07)*, pp. 584–589, Mar. 2007.

[12] D. Coeurjolly, S. Miguet, and L. Tougne, “Discrete curvature based on osculating circle estimation,” *Visual Form 2001*, pp. 303–312, 2001.

[13] J. H. Ahlberg, E. N. Nilson, and J. L. Walsh, *The theory of splines and their applications*. New York: Academic Press, 1967.

[14] M. Hazewinkel, Ed., *Encyclopaedia of mathematics: an updated and annotated translation of the Soviet ”Mathematical encyclopaedia”*. Dordrecht ; Boston : Norwell, MA, U.S.A: Reidel ; Sold and distributed in the U.S.A. and Canada by Kluwer Academic Publishers, 1988.

[15] [Online]. Available: http://www.siepic.ubc.ca/GSiP

[16] F. Pikus, Z. Lu, and P. Brooks, “Programmable electrical rule checking,” Dec. 2 2010, uS Patent App. 12/474,240.