CROSSING-AWARE CHANNEL ROUTING FOR PHOTONIC WAVEGUIDES

Christopher Condrat, Priyank Kalla, Steve Blair
Department of Electrical and Computer Engineering
University of Utah, Salt Lake City, UT, USA

***Abstract*—Silicon photonics technology is progressing at a rapid pace. Despite greatly expanded manufacturing capability, physical design of integrated optical circuits currently lacks the level of automation found in VLSI design. A key component of integrated optic design is waveguide routing; however, unlike VLSI, where signal nets are routed with metal layers and vias, photonic waveguides are fabricated in *planar* substrates. For many applications, our studies show that the waveguide routing problem can be formulated as *planar channel routing*. Signal losses become a major factor due to the insertion losses of planar waveguide crossings. Channel routing must therefore take into account these losses. This paper investigates methods for adapting traditional VLSI channel routing techniques for integrated optics — specifically, a technique based on left-edge-style track assignment. We show how incorporating waveguide crossing constraints into the underlying constraint model affects the routing solution, and describe the necessary modifications and extensions to the routing technique to properly exploit optical technology. We implement the channel router, describe the experimental results, and compare the cost of solutions with respect to waveguide crossings, corresponding to signal loss, and channel height.**

1. INTRODUCTION

The growth of silicon photonics technology necessitates the development of design automation techniques in order to fully take advantage of large scale integration. The Electronic Design Automation (EDA) community is therefore investigating how automatic design space exploration techniques can be adapted to the photonics domain [1]–[4]. Automated waveguide routing will play an important part in large-scale optical network design, enabling applications that require technology scaling far beyond that of traditional optical systems.

This paper presents a methodology and solution for routing of integrated optical waveguides. In particular, we show that the detailed routing problem manifests itself as a *channel routing* problem [5], where optical waveguides are fabricated on a planar substrate and are connected to terminals at the ends of the channel. Traditional channel routing techniques can be applied to routing this channel; however, the single-layer planar nature of the optical substrate requires that the constraint models be suitably modified from their VLSI origins.

Formulation and objectives: An electro-optic circuit consists of a set of pre-designed optical devices — modulators, couplers and splitters, detectors, etc. — fabricated on a planar substrate, and interconnected with waveguides. This work is concerned with the routing of these interconnecting waveguides, and not with device placement, which is performed using VLSI floorplanning tools, specifically Capo [6].

The overall electro-optic circuit synthesis framework is depicted in Fig.1. Using a placement tool, pre-designed optical devices (a) are placed in fixed-width columns (b). Such a placement gives rise to **vertical routing channels** (c), which are routing regions that separate the placed devices. Waveguides are routed between devices at "ports" (d) that face the channels. For ports in different columns, these waveguides may pass through **horizontal routing channels**, as depicted in (e). Waveguides are fabricated on planar substrates; however, they may cross each other perpendicularly (f) at the cost of a small amount of signal loss.

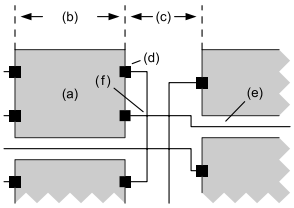


Fig. 1: Optical channel routing formulation

The waveguide connecting two ports is denoted as a "net" and comprises a single route with no signal sharing (fanout). Signal sharing does not take place within the routing region, but rather is explicitly performed by pre-designed waveguide splitters in the device columns. Our methodology therefore renders every net a **two-terminal net** within the channel.

A key difference between traditional semiconductor routing and optical waveguide routing is in the *planar* nature of the optical substrate, permitting no transitions to other layers using vias. Waveguides can cross; however, they can only cross perpendicular to each other using specially designed crossings [7], [8]. Losses are on order of *0.1-0.2dB* ($3.5-4.5\%$) *per crossing.*

The primary optimization objective in our routing formulation is the **minimization of the total number of waveguide crossings within the channel**. Though bend-loss [9] is an important loss mechanism in integrated waveguide systems, the number of bends in a channel routed by left-edge style channel routers is a fixed quantity due to a set number of transitions from vertical to horizontal spans; therefore, the number bends cannot be minimized further. Bulk straight waveguides have negligible losses (dB/cm) at the scales this router operates at. Though channel height minimization is the goal of traditional VLSI channel routers, signal loss is more important for optical applications; therefore channel height reduction is the secondary objective of this technique.

Previous work: In VLSI physical design, channel routing algorithms are textbook knowledge [5], [10]; our work is based on left-edge-style channel routers [11], [12]. The primary objective of such routers is to minimize track height, not avoid crossings. Investigations have been made into via minimization in channel routing [13]. Vias enable nets to change routing layers in order to cross other nets. However, in channel routing, vias correspond to transition points between horizontal and vertical routing spans — the equivalent of *bends* in a channel routing, not crossings. Crossing minimization for

This research was funded in part by a sub-contract to AFOSR grant FA9550-09-1-0661.

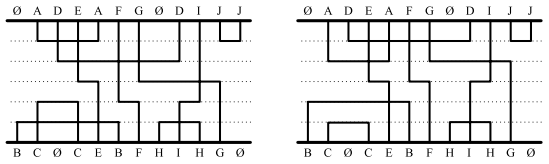
978-1-4799-0066-4/13/$31.00 ©2013 IEEE

channel routing has been studied in the context of QCA routing [14]. Track assignments for multi-terminal nets induce varying numbers of crossings, and [14] formulates crossing minimization, heuristically, as a weighted-minimum-feedback-edge-set problem. However, in the context of our problem — utilizing exclusively 2-terminal nets — we eliminate crossings altogether when crossings can be avoided.

We investigate methods for adapting traditional VLSI channel routing techniques for integrated optics, describe our new constraint models and how these are applied to the channel routing technique, and present a channel router suitable for optical waveguide routing.

2. LEFT-EDGE-STYLE CHANNEL ROUTING

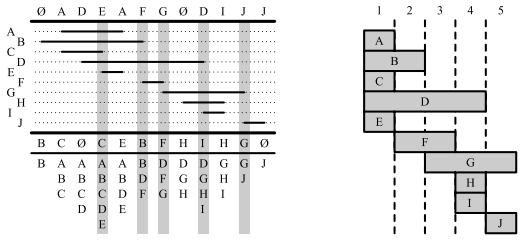
This paper uses the standard horizontal convention for representing channels, as depicted in Fig.2(a), with net signal *pins* residing on the top and bottom of the channel at fixed column locations; signals denoted with "Ø" are empty terminals not connected to any net. The channel routing area is the rectangular region between the pins on the top and bottom. To perform routing, vertical connections are made to horizontal spans located at fixed vertical positions, denoted *tracks*. The number of tracks needed to route all nets in a channel is the *height* of the channel. Traditional channel routing seeks to minimizing the height of a fully routed channel.



(a) Track-optimized (5 tracks, 10 crossings). (b) Crossing-constrained (5 tracks, 8 crossings).

Fig. 2: Channel routing solutions under differing constraints.

Channel routing formulation: A channel problem can be represented using horizontal and vertical constraint graphs (HCG, VCG). An alternate representation of the HCG is the zone representation, which is derived from the HCG, where every zone is defined by a maximal clique. The number of signals in the largest zone is the lower bound on the number of tracks needed for routing. These graphs encode constraints on how tracks may be assigned to nets in the channel. Consider the channel routing problem depicted in Fig.3(a). The resulting zone representation is depicted in Fig.3(b). Likewise, the VCG for the problem is represented in Fig.5(a).



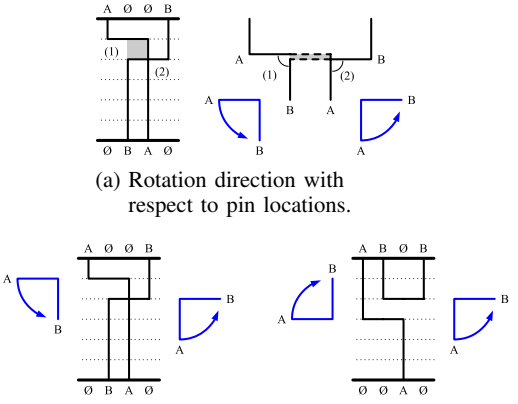
(a) Five maximal subsets of signals. (b) Resulting five zones.

Fig. 3: Horizontal constraints and zone representation.

A net may be assigned to a track should it have no descendants on the VCG, and have no overlapping zone conflicts with previously assigned nets on a given track. Nets are removed from the VCG as they are assigned to tracks. When a track cannot contain more nets, a new track is created and the process is repeated until no more nets are left for assignment.

Multiple nets can be candidates for assignment to a given track, each with different horizontal overlaps. Therefore heuristics are used to choose which nets are assigned first. One of the simplest is a greedy heuristic used in *constrained left-edge channel routing* [15], where the left-most available nets in channel are assigned first to tracks. This can lead to sub-optimal track-utilization; more sophisticated heuristics analyze the graph structure for better results, such as [12], which attempts to reduce the longest path in the VCG for better track utilization. We refer to the class of track assignment algorithms above generically as "left-edge-style" channel routing. The approach we describe below can be incorporated into any such techniques.

Crossing-constrained track assignment: Fig.2 depicts two different solutions to a channel routing problem.Fig.2(a) depicts the output of a left-edge 2-layer channel router, and Fig.2(b), a channel routing constrained for crossing minimization. Both solutions are minimal in terms of tracks; however, the total number of crossings in Fig.2(a) is 10, compared to 8 in Fig.2(b). The discrepancy in the number of crossings is attributed to the two crossing points caused by nets *B* and *C*. By forcing *C* to appear below *B*, two crossings are avoided. However, transforming from Fig.2(a) to Fig.2(b) is not as simple as moving net *C* below *B*, not if track height is to be kept minimal. Crossing minimization must therefore be encoded into the routing process itself as constraints.



(a) Rotation direction with respect to pin locations.

(b) Same rotation direction ⇒ Unavoidable crossing. (c) Opposite rotation directions ⇒ Avoidable crossing.

Fig. 4: Crossing detection via rotation from *A* to *B*.

We constrain the channel routing problem to favor crossing minimization. The VCG is modified such that avoidable crossings impose vertical constraints on the net ordering. Only nets that share zones have the possibility of crossing, and pairwise analysis takes place after the zones are derived.

A crossing constraint is only encoded into the VCG if a crossing can be avoided. For example, the pair of nets in Fig.4(c) would not normally be constrained in the VCG; however, a net crossing can be avoided if *B* is assigned a track above *A*. Therefore, an edge connecting *B* to *A* is added

to the VCG. Conversely, the two nets in Fig.4(b) cannot avoid crossing, and therefore no constraint is added.

We introduce the concept of pin *rotation* to detect avoidable crossings. If we were to map the pins of nets on a unit circle, a crossing is unavoidable if rotating from one pin to the next is not possible without first passing through the pin of another net. Consider the nets depicted in Fig.4(a). Collapsing the shared horizontal region, and considering the areas Fig.4(a)(1) and Fig.4(a)(2) shows how pins of a given side rotate with respect to each other (clockwise/counter-clockwise) around an axis fixed at the center. In the case of Fig.4(a)(1), the rotation of the left pin of A to the left pin of B is *counterclockwise,* and likewise the pins on the right-side also rotate in the same *counterclockwise* direction. If the pins on both left and right terminals rotate in the same direction a crossing is unavoidable. More formally:

$$X^{left}_{A,B,CW} = \begin{cases} X^{left}_{B,top} & \text{if } C^{left}_{A} < C^{left}_{B} \\ \neg X^{left}_{A,top} & \text{otherwise} \end{cases} \quad (1)$$

$$X^{right}_{A,B,CW} = \begin{cases} X^{right}_{A,top} & \text{if } C^{right}_{A} < C^{right}_{B} \\ \neg X^{right}_{B,top} & \text{otherwise} \end{cases} \quad (2)$$

$$X_{\text{avoidable}}(A,B) = \left(X^{left}_{A,B,CW} \neq X^{right}_{A,B,CW} \right) \quad (3)$$

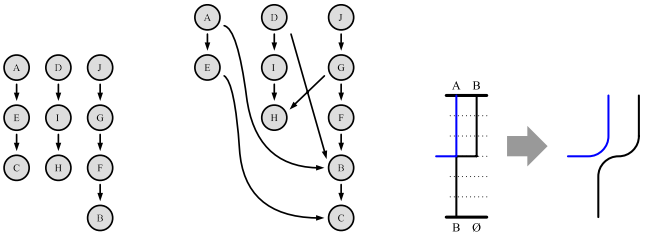
where $C^{left/right}_{N}$ is the integer-valued column-position of a pin of net N on a given side (left, right); the Boolean variable $X^{left/right}_{N,top}$, using the same notation, denotes whether that pin resides on the top side of the channel. Eqn.1 and Eqn.2 utilize the horizontal relationships of pins and their channel-sides (top/bottom) to determine the *clockwise rotation* (CW) of a given pair of $left$ or *right* pins for nets A and B, rotating from A to B. A crossing is avoidable only if left and right rotations are *not* the same, the result of Eqn.3.

For example, in Fig.4(a), consider the left side of the shared span Fig.4(a)(1):

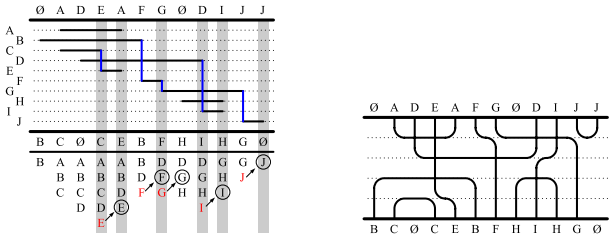
- The variables C^{left}_{A} and C^{left}_{B} are the column positions of the respective left-terminals of nets A and B. In the example, $C^{left}_{A} = 1$, $C^{left}_{B} = 2$.
- $C^{left}_{A} < C^{left}_{B}$ implies $X^{left}_{A,B,CW} = X^{left}_{B,top}$ from Eqn.1.
- The left pin of net B is *not* on the top side of the channel ($X^{left}_{B,top} = false$). Therefore, the left side of the pair of nets is *not* rotating clockwise from A to B, i.e. $X^{left}_{A,B,CW} = X^{left}_{B,top} = false$.
- On the right side of the shared span Fig.4(a)(2), $C^{right}_{A} < C^{right}_{B}$. This condition implies that $X^{right}_{A,B,CW} = X^{right}_{A,top} = false$. The right side is therefore also *not* rotating clockwise from A to B.

Having the same direction of rotation ($X^{left}_{A,B,CW} = X^{right}_{A,B,CW} = false$) implies that a crossing is *unavoidable,* as determined by Eqn.3; this is reflected in the figure.

Applying crossing constraints to the problem depicted in Fig.3(a) results in the VCG depicted Fig.5(b). As compared to the original VCG Fig.5(a), the crossing-constrained VCG is more heavily constrained, ensuring that unnecessary crossings do not occur, such as the double-crossing of nets B and C in Fig.2(a).



(a) Original VCG. (b) With crossing constraints. (c) Knock-knee implementation.



(d) Knock-knee-constrained zone representation. (e) 4-track routing solution utilizing knock-knees.

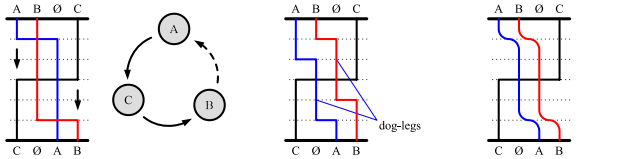
Fig. 5: VCGs for Fig.3(a) and knock-knee extension.

Knock-knee track sharing: Though the modified VCG is effective in preventing waveguide crossings, the additional constraints can affect overall track height, and may produce a worse solution in terms of number of tracks. However, we observe that the bend geometry of optical waveguides can be exploited to further reduce channel height. This is discussed below.

Consider the two nets in Fig.5(c). The endpoints of the two nets occupy the same column and therefore net A should be placed above B in the VCG. However, given the same track, the two nets would intersect at a corner of each horizontal span–a *knock-knee*. In VLSI, this situation is untenable, and different tracks would need to be assigned to each net. However, for waveguides, the minimum grid spacing for a channel can permit knock-knees in the routing grid. This is depicted in Fig.5(c), where a track is shared between the two nets without overlap.

A knock-knee occurs where one net ends and another begins, e.g. nets C and E in Fig.5(e). During zone construction, at columns where knock-knees appear, the net that is beginning its horizontal span is only added to the *subsequent* column set, rather than the current column set under consideration. For example, in Fig.5(e) knock-knee signals E, F, G, I, and J are removed from the marked columns and only appear in the subsequent columns. The effect of this column change on the resulting zones is demonstrated in Fig.5(d), where there are six (6) zones rather than the five (5) from the previous zone analysis Fig.3. Despite containing an additional zone, the largest column set now contains *one fewer* net than the original, resulting in the 4-track solution depicted in Fig.5(e).

Overall, the effect incorporating knock-knees into a routing solution is that two knock-knee nets can now occupy separate zones, and therefore can be placed on the same track. Additional zones may be created; however, those zones are equal in size or *smaller* in terms of nets — *potentially reducing the lower bound on the number of tracks required for routing*.



(a) Vertical cyclic constraints. (b) Dog-legging avoids crossings. (c) Knock-knees avoid additional tracks.

Fig. 6: Cycles induced by crossing constraints.

Cycles induced by crossing constraints: Crossing constraints can induce cycles in the VCG. Consider the three nets depicted in Fig.6(a). Without crossing constraints, nets *A* and *B* would be unconstrained, and no cycle would occur; however, due to the constraint edge between *B* and *A* such a cycle occurs. Cyclic constraints cannot be routed without additional tracks and require "doglegging" to complete routing [11]. In order to avoid crossings, the routes for *A* and *B* are converted into doglegging routes as depicted in Fig.6(b), utilizing the same columns as the original. Unfortunately, this results in an additional two (2) tracks being added to the routing solution should spare tracks not be available adjacent to the cycle. However, in the presence of knock-knees, both the crossings, and the additional tracks can be avoided, as depicted in Fig.6(c). The experimental results show that knock-knees can have a marked difference in track utilization especially in the presence of cyclic constraints induced by crossings.

TABLE I - **YK** = Yoshimura-Kuh router; **CA** = Crossing-aware router, **W** = channel width; **B**, **T** = # of bottom, top-only nets

					Crossings		Tracks	
Design	**Nets**	**W**	**B**	**T**	**YK**	**CA**	**YK**	**CA**
C16.0	17	17	5	5	57	39	10	11
C16.1	15	18	5	3	52	38	10	11
C16.2	16	22	3	3	62	50	11	13
C16.3	15	24	3	3	46	28	10	12
C16.4	19	34	3	3	106	62	11	14
C32.0	33	33	11	11	219	137	15	19
C32.1	30	36	7	7	253	151	18	21
C32.2	31	43	7	7	275	199	21	23
C32.3	34	54	6	6	429	247	24	24
C32.4	35	63	5	7	396	254	20	24
C64.0	62	62	20	20	781	497	32	39
C64.1	64	76	16	16	1 204	726	39	49
C64.2	62	86	15	15	1 111	635	35	43
C64.3	72	115	14	14	1 593	977	38	54
C64.4	70	126	11	11	1 363	923	38	49
C128.0	135	135	45	45	4 854	2 544	76	98
C128.1	114	136	38	38	3 380	1 754	63	81
C128.2	107	149	21	26	3 703	2 203	64	83
C128.3	103	164	20	20	3 038	1 872	58	80
C128.4	127	228	25	21	5 031	3 303	69	90
C256.0	212	212	70	70	10 608	5 870	105	141
C256.1	275	330	68	68	19 901	13 175	148	201
C256.2	256	358	64	64	17 943	10 719	134	176
C256.3	237	379	47	47	15 877	10 255	133	173
C256.4	274	493	45	45	23 036	14 670	154	214

3. Experimental Results

We compare the crossing-aware (CA) left-edge style router against the Yoshimura-Kuh (YK) left-edge-style router [12] on a number of channel routing instances. We choose the YK channel router as the prototypical VLSI left-edge-style router incorporating efficient graph-based heuristics for track-minimization. Our routing instances have varying numbers of nets, and widths, as well as the number of nets with both pins on the same side (i.e. top/bottom-only nets) in order to diversify net pin connections. Routing solutions are evaluated in terms of crossings as a primary metric, as well in number of tracks.

The results of the experiments are presented in Table I. Both routers completed their routing in under 10 seconds for all channel instances. On average, the CA reduces the number of crossings to 0.607× of those produce by the YK router; this comes at a cost of roughly 1.25× the number of tracks. The number of nets had the most significant effect on both crossings and track counts, with the greatest benefits in terms of crossings versus tracks obtained with fewer overall nets (e.g. 16 and 32). Other factors such as channel width, and number of top/bottom-only nets having little effect.

4. Conclusion

This paper describes an investigation into channel routing for integrated optics. Our studies show that optical constraints, specifically waveguide crossing constraints, must be incorporated into the channel routing models. Our investigations into left-edge-style channel routers show that by incorporating crossing-minimization constraints, tracks can be assigned to reduce crossings during routing. These constraints are incorporated into our crossing-aware router, and we also improve track utilization by incorporating waveguide curvature into the routing model–specifically allowing knock-knees during track assignment.

Our experiments show that the number of crossings produced by our router are substantially reduced compared to other left-edge-style routers such as [12]. While this crossing-minimization comes at the cost of additional tracks, we feel this is a reasonable trade-off, especially at lower net counts. Overall, this channel router reduces signal losses due to crossings in an automated routing framework, an important part of photonic waveguide integration.

References

[1] D. Ding and D. Pan, "Oil: A nano-photonics optical interconnect library for a new photonic network architecture", *in System-Level Interconnect Prediction Workshop (SLIP)*, 2009.

[2] D. Ding, Y. Zhang, H. Huang, R. T. Chen, and D. Z. Pan, "O-Router: an optical routing framework for low power on-chip silicon nano-photonic integration", *in Design Auto. Conf.*, pp. 264–269, 2009.

[3] J. Orcutt and R. Ram, "Photonic Device Layout within the Foundry CMOS Design Environment", *IEEE Photonics Tech. Lett.*, 2010.

[4] C. Condrat, P. Kalla, and S. Blair, "Logic Synthesis for Integrated Optics", *Proc. ACM Great Lakes Symp. on VLSI*, pp. 13 – 18, 2011.

[5] A. B. Kahng, J. Lienig, I. L. Markov, and J. Hu, *VLSI Physical Design: from Graph Partitioning to Timing Closure*, Springer , 2010.

[6] J. Roy et al., "Capo: robust and scalable open-source min-cut floorplacer", *in Proc. 2005 Int Symp on Physical Design*, ISPD '05, pp. 224–226, New York, NY, USA, 2005. ACM.

[7] W. Bogaerts, P. Dumon, D. V. Thourhout, and R. Baets, "Low-loss, low-cross-talk crossings for silicon-on-insulator nanophotonic waveguides", *Opt. Lett.*, vol. 32, pp. 2801–2803, Oct 2007.

[8] F. Xu and A. W. Poon, "Silicon cross-connect filters using microring resonator coupled multimode-interference-based waveguide crossings", *Opt. Express*, vol. 16, pp. 8649–8657, Jun 2008.

[9] Y. Vlasov et al., "Losses in single-mode silicon-on-insulator strip waveguides and bends", *Opt. Express*, vol. 12, pp. 1622–1631, 2004.

[10] S. Sait et al., *VLSI Physical Design Automation*, IEEE Press, 1995.

[11] D. Deutsch, "A Dogleg channel router", *in Proc. 13th Design Automation Conference*, pp. 425–433, New York, NY, USA, 1976. ACM.

[12] T. Yoshimura and E.S. Kuh, "Efficient Algorithms for Channel Routing", *IEEE Trans. Computer-Aided Design of Integrated Circuits and Systems*, vol. 1, pp. 25–35, 1982.

[13] C. Cheng and D.N. Deutsch, "Improved channel routing by via minimization and shifting", *in Design Automation Conference, 1988. Proceedings., 25th ACM/IEEE*, pp. 677–680, 1988.

[14] S. B. Stephen and L. S. Kyu, "QCA channel routing with wire crossing minimization", *in Proc.15th ACM Great Lakes Symp. on VLSI*, GLSVLSI '05, pp. 217–220, New York, NY, USA, 2005. ACM.

[15] A. Hashimoto et al., "Wire routing by optimizing channel assignment within large apertures", *in Proc. 8th Design Automation Workshop*, pp. 155–169, New York, NY, USA, 1971. ACM.