Constraints-aware Adaptive Routing with Hybrid Waveguides for Photonic Integrated Circuits

Yuchao Wu[1,2*], Xiaofei Yu[1*], Xianyi Feng[1], Yeyu Tong[1,2], Yuzhe Ma[1†]
[1]Microelectronics Thrust, HKUST(GZ)
[2]Guangdong-Macao Joint Laboratory for Modular Chip Design and Testing, HKUST(GZ)
{ywu092,xyu082,xfeng262}@connect.hkust-gz.edu.cn {yeyutong,yuzhema}@hkust-gz.edu.cn

2025 IEEE/ACM International Conference On Computer Aided Design (ICCAD) | 979-8-3315-1560-7/25/$31.00 ©2025 IEEE | DOI: 10.1109/ICCAD66269.2025.11240866

***Abstract*—Photonic integrated circuits (PICs) have emerged as a promising solution for next-generation computing and interconnects, delivering enhanced performance in bandwidth and energy efficiency compared to conventional electronics. Previous studies on automatic optical routing have typically assumed uniform waveguide classes, neglecting the real-world requirement for multiple waveguide types to enhance performance and integration density. To address this gap, we propose an adaptive waveguide routing framework explicitly tailored for PICs that effectively handles hybrid waveguide classes. Our method starts with a fast initial routing algorithm that optimizes waveguide paths, considering target net lengths or length matching constraints among nets, minimizing bends, and ensuring compatibility with hybrid waveguide usage. We further enhance routing quality using a mixed integer linear programming (MILP)-based automatic transition insertion technique, strategically employing different waveguide types to significantly reduce transmission loss and ensure compliance with stringent matching constraints. Experimental results show that, compared to prior approaches, our algorithm reduces total insertion loss by up to 37% on average in target-length routing scenarios, and achieves an average reduction of 5% in total insertion loss and 14% in maximum loss in multi-net cases while satisfying all constraints.**

I. Introduction

As Moore's Law approaches its physical limitations, traditional electronic interconnects are increasingly challenged to meet the escalating demands of artificial intelligence (AI) applications. Photonic integrated circuits (PICs) have emerged as a compelling alternative to traditional electronic circuits and represent a promising path for sustaining continued computational progress [1]–[5].

To achieve enhanced performance and integration density, PIC designers intentionally combine multiple classes of waveguides, such as ridge, rib, or even buried waveguides, each offering specific advantages in terms of propagation loss, bend tolerance, and layout footprint [6]. The hybrid waveguide layout helps reduce insertion loss and enables tighter bends in congested regions, thereby improving overall PIC performance and integration density. Fig. 1 illustrates the cross-sections and features of three typical waveguides fabricated in the same fabrication cycle through varying etch depths and the transitions between them. As shown, stronger optical confinement, such as in a ridge waveguide, enables a smaller bend radius but typically results in higher propagation loss due to sidewall roughness, whereas shallow-etched structures exhibit lower loss at the cost of reduced bend tolerance. Due to the optical fields, effective indices, and mode sizes differ, each transition between waveguide types introduces its own excess loss and phase shift, so an adaptive hybrid layout must carefully manage where and how many such transitions occur.

Independent of waveguide choice, optical routing is governed by a distinct set of constraints compared to its electrical counterpart, due to the inherent differences between waveguides and metal wires. In contrast to electrical interconnects, photonic waveguides must satisfy stringent geometric constraints, such as minimum bend radius and spacing requirements, while still allowing waveguide crossings within

*Equal contribution
†Corresponding author

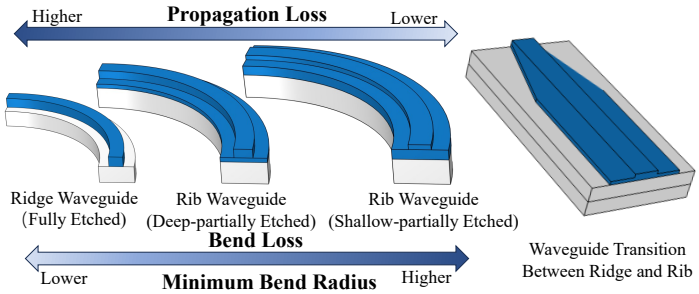

Fig. 1 Illustration of three typical silicon photonic waveguide types, their characteristics, and the transitions.

the same layer [7]. Beyond geometry, the path-length matching constraint is critical wherever coherent interference is exploited, such as Mach–Zehnder interferometers [8] and ring resonator–based filters [9], because even sub-micron mismatches translate into appreciable phase error at near-infrared wavelengths. Complementing phase balance, the loss matching constraint requires that parallel paths maintain equal or tightly constrained differential attenuation to preserve amplitude symmetry and extinction-ratio specifications [10]. Certain subsystems, such as pulse-reshaping circuits and optical phased-array feeds, also impose the absolute path-length constraint. These systems often permit only a few percent deviation from a prescribed effective length to achieve the desired group-delay profile or timing alignment [11]. These geometric, length, and loss constraints, along with discrete waveguide class choices, form the core challenges of **constraint-aware routing with hybrid waveguides**. Consequently, it is desirable to develop unified PIC routing techniques capable of addressing these issues that minimize total insertion loss while adhering to all complicated constraints. However, developing such a design flow often demands considerable expertise and hands-on effort, particularly when addressing complex layouts and optimizing performance [7], [12].

Recent research has extensively focused on optical routing problems. Ding *et al*. [13] developed the first gridless optical routing framework based on integer linear programming (ILP) for low-power on-chip optical interconnections. To ensure the correct functionality, Chuang *et al*. [14] first considered the matching constraints and developed an ILP-based optical routing framework that optimizes the total insertion loss while satisfying all matching constraints. Subsequently, Wu *et al*. [15] proposed a fast diffuse-based matching algorithm combined with an initial routing strategy based on A*, which effectively reduces overall loss while maintaining compliance with matching constraints.

Notwithstanding the abundant studies, all of the previous work has typically assumed uniform waveguide classes. However, practical PIC designs often employ multiple classes of waveguides, each governed by distinct design rules, which remain relatively underexplored in the literature. Related efforts exist in other routing domains: Du *et al*. [16] proposed a multi-level maze routing algorithm to address analog

routing scenarios involving nets with variable wire widths. Liu *et al.* [17] incorporated a neck mode for differential pairs in PCB routing that adaptively adjusts line width and spacing to navigate through routing channels. However, beyond the fundamental differences between optical and electrical routing noted earlier, these approaches only consider nets with different predefined widths or spacings, which makes them inadequate for addressing the adaptive waveguide routing problem in PIC.

In this paper, we propose the first complete optical routing framework that effectively handles hybrid waveguide classes, which can minimize the total insertion loss while satisfying all constraints. Our method begins with an efficient algorithm for initial routing that optimizes waveguide paths by considering target net lengths or path length matching constraints among nets, while also minimizing bends and avoiding unnecessary short segment fragmentation. This is followed by an adaptive refinement stage that strategically assigns waveguide types using a mixed-integer-linear programming (MILP)-based formulation to reduce total insertion loss while ensuring compliance with key design constraints.

Our contributions are summarized as follows:

- We propose a complete optical routing framework that minimizes overall insertion loss by considering multiple types of waveguides while rigorously satisfying all matching constraints. To the best of our knowledge, this is the first optical router capable of effectively handling hybrid waveguide classes.
- We introduce an efficient diffuse-based routing algorithm that accounts for the target net length while reducing the number of bends and minimizing the creation of unnecessary short segments.
- We present an MILP-based automatic transition insertion technique tailored for the optical routing problem with matching constraints. This formulation further reduces total loss by leveraging the properties of different waveguide classes while satisfying all matching constraints.
- Experimental results show that, compared to prior approaches, our algorithm reduces total insertion loss by up to 37% on average in target-length routing scenarios, and achieves an average reduction of 5% in total insertion loss and 14% in maximum loss in multi-net cases. Additionally, our algorithm delivers a 250× speedup over [18], completing all designs within seconds while rigorously satisfying all constraints across all testcases.

II. Preliminaries

In this session, we first give a brief introduction to the photonic design rules and outline the rules related to adaptive waveguides. Then, we further introduce the losses used in our work. Finally, we present the formulation of the adaptive waveguide routing problem.

A. Photonic Design Rules

The routing of PICs requires a set of specific design rules to ensure manufacturability and high signal integrity. Specifically, photonic waveguides demand: (1) minimum spacing to prevent crosstalk, (2) precise port alignment to ensure low-loss connections. In addition to these fundamental constraints, resolving our problem necessitates us to focus on the following three key design rules:

1) *Bend Radius*
The bend radius is a critical factor in PIC design, particularly when routing waveguides around corners or obstacles. Unlike the 90° metal wire bends in VLSI, photonic waveguides require smooth, gradual curves to maintain light confinement and minimize losses. Common bend types include circular, Euler-shaped, and sine-shaped, while sharp bends with radii smaller than the recommended minimum are avoided to prevent signal degradation. The bend radius is affected by factors such as waveguide material, geometry, and refractive index contrast. For example, with deeper etching, ridge waveguides allow smaller bend radii, while rib waveguides require larger radii to avoid large losses.
2) *Waveguide Transition*
A waveguide transition is a critical component that enables signal transmission between different types of waveguides. It can enhance the circuit performance and routability by leveraging the distinct characteristics of various waveguides, such as bend loss, propagation loss, and bend radius. However, each transition introduces insertion loss and occupies additional space. In this paper, the insertion loss ranges from 0.5 dB to 1.7 dB, and the transition footprint is approximately $20 \times 11 \mu\text{m}^2$. Consequently, integrating hybrid waveguides necessitates considering both the properties and spacing.
3) *Matching Constraints*
Matching constraints play a crucial role in signal integrity, as well-matched circuits are capable of preserving their amplitude, phase, and timing accuracy, thus ensuring a high signal-to-noise ratio (SNR). Achieving this requires both length matching and loss matching. Length matching minimizes delay mismatches between optical paths, thereby maintaining phase coherence and preventing functional issues caused by timing misalignment. In contrast, loss matching ensures a balanced power attenuation, thereby preventing SNR degradation.

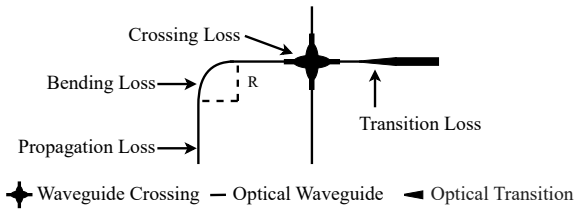

Fig. 2 Illustration of three major sources of loss for optical routing.

B. Insertion Loss

The total insertion loss L_{total} is primarily determined by four sources: crossing loss L_{cross}, bending loss L_{bend}, propagation loss L_{prop}, and transition loss L_{tran}, as illustrated in Fig. 2. It can be expressed as:

$$L_{total} = L_{cross} + L_{bend} + L_{prop} + L_{tran}. \quad (1)$$

C. Problem Formulation

The constraints-aware adaptive waveguide routing problem can be formally defined as follows:

Given a netlist, a set of pins, a set of waveguide types, a group of length-matching nets, a group of loss-matching nets, and the design rules, route all nets and utilize distinct waveguides to minimize the total insertion loss such that no design rule is violated, and all the nets within the length-matching and loss-matching groups satisfy their respective matching constraints.

III. Methods

In this section, we first provide an overview of our routing flow and then introduce the details.

A. Algorithm Overview

Fig. 3 summarizes our algorithm flow, which consists of four stages: (1) preprocessing, (2) initial routing, (3) automatic transition insertion, and (4) postprocessing.

During preprocessing, square routing grids are constructed, and the routing area is partitioned based on obstacle distribution. The initial routing is then applied to establish preliminary paths, and an efficient diffuse-based method is applied to transform each single-net route into

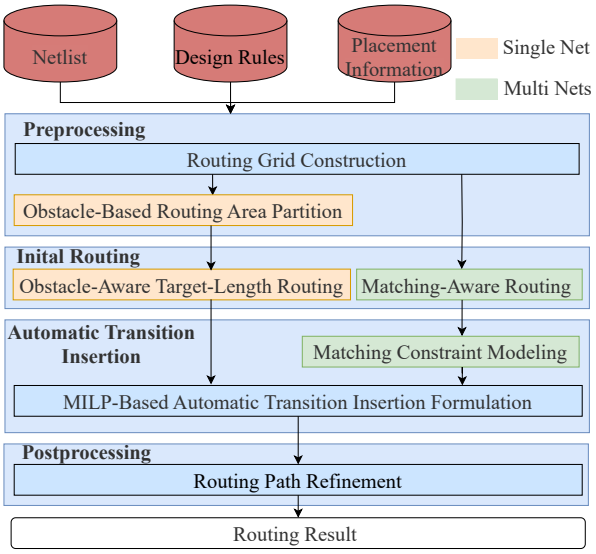

Fig. 3 The overview of our proposed routing flow

a path that meets its target length while minimizing total insertion loss and ensuring compatibility with automatic transition insertion. A modified diffuse-based method, adapted from [15] and extended to consider adaptive waveguide routing, is used to generate length-matching results with minimal loss across the group of matching nets. Based on the initial routing result, an MILP formulation is used to perform automatic transition insertion, minimizing insertion loss while also satisfying loss-matching constraints among the net groups. Finally, in the postprocessing stage, the positions of pins and bend points are refined while considering matching constraints to obtain the actual physical interconnections.

B. Preprocessing

1) Grid Construction

We adopt a square grid structure for the routing process, with a grid size determined by the minimum bend radius of the routed waveguide. Although the distinct available waveguide types exhibit distinct minimum bend radius requirements, we default to the waveguide class with the smallest minimum bend radius for initial routing. Additionally, the capacity of each edge is set to only one to avoid the potential crossings that could occur between actual curved bends in the physical layout. These design choices maximize routability and ensure that all routes adhere to fabrication design rules and remain physically realizable without layout violations.

2) Obstacle-Based Routing Area Partition

Following grid construction, we extract all obstacle boundaries, including existing circuit components and fixed placement blocks. From each horizontal and vertical edge of these obstacles, we extend lines across the layout domain in the respective direction. These extended lines intersect with the layout's outer boundary, resulting in a partitioning of the routing space into axis-aligned rectangular subregions.

Fig. 4 demonstrates a simple example of our routing area partition: Fig. 4(a) illustrates an actual routing scenario containing obstacles and routing paths, while Fig. 4(b) presents the corresponding partitioned layout. We further classify the separate layout regions into several categories based on connectivity and boundary enclosure, and mark them in different colors. Specifically, the red-shaded areas denote the *dead regions*, characterized by being enclosed on three sides by either obstacles or the layout boundary, with only one routing capacity remaining on the fourth side. Due to limited accessibility, these regions do not allow bidirectional passage, making detouring through them

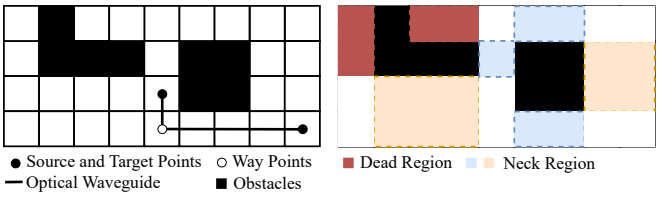

Fig. 4 Region partition and classification.

impossible and routing infeasible. As a result, they are marked as unreachable and excluded from routing unless they contain a start or end terminal. In contrast, the yellow and blue regions depict the *neck regions*, whose opposing boundaries are constrained either by layout edges or by adjacent obstacles. These regions, though partially accessible, have significantly restricted routing potential because their capacity for path expansion is inherently limited by surrounding structures. Within this category, the blue-colored regions represent a particularly constrained scenario, where the routing capacity on the accessible side is merely one. This spatial partitioning effectively captures the structure imposed by obstacles and delineates clear routing corridors. The strategic utilization of these subregions, particularly their influence on routing decisions, is elaborated further in the Section III-C1.

C. Constraints-Aware Initial Routing

1) Single-Net Routing Under Target Length Constraints

Absolute path-length constraints commonly occur in practical PIC designs. Subsystems such as dispersion-engineered delay lines, pulse-shaping circuits, and optical phased-array feeds typically require the optical path to remain within a few percent of a specified target length to ensure precise group-delay characteristics and maintain accurate timing alignment. To address this challenge, our algorithm implements a diffusion-based routing method that efficiently satisfies stringent length constraints. The approach not only minimizes optical insertion loss by reducing unnecessary bends but also ensures compatibility with subsequent automatic transition insertions. Our method involves an initial maze routing connection, followed by a structured detour phase to achieve target lengths. This process incorporates adaptive strategies that navigate regions with limited spatial availability, ensuring effective routing under diverse layout conditions.

Definition and Implementation of Detour Units: A detour unit is defined as a square or rectangular expansion pattern whose width corresponds to the bend radius requirements specific to different waveguide classes. As described in Section III-B1, the routing grid is constructed based on the smallest minimum bend radius among all waveguide types. The width of each detour unit is defined as a multiple of this minimum value, rounded up to the nearest integer. For example, waveguides with minimum bend radii of $10\mu m$, $15\mu m$, and $40\mu m$ correspond to detour units with widths of 1, 2, and 4 grid cells, respectively. The region covered by a detour unit but not occupied by the detour lines is marked as a low-priority region, which is reserved and not used in the initial phase, in order to prevent the formation of early bends with small detour lengths.

Diffuse-based Routing Strategy: The proposed routing methodology begins with an initial connection established between the source and target points, utilizing a conventional maze routing algorithm [19]. After securing this preliminary route, a detour phase is initiated to incrementally extend the optical path length to the desired target value. This detouring phase strategically inserts additional routing segments to extend the path, while carefully minimizing optical insertion loss and maintaining compatibility with future automatic transition insertion.

Fig. 5 demonstrates this process in detail. In Step 1, the source and

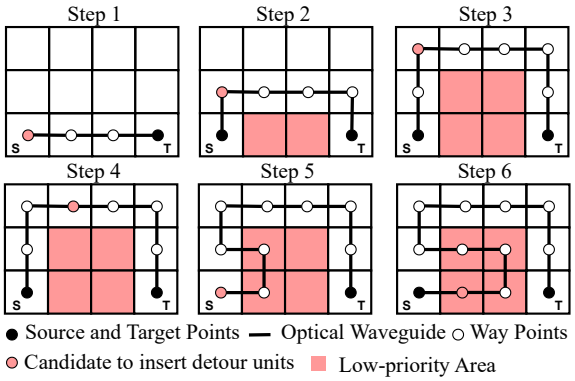

Fig. 5 Illustration of the target length routing process.

target points are connected using a conventional maze routing algorithm, and the source point is selected as the initial candidate for detouring, as indicated by the shaded red points. At each candidate location, the algorithm assesses the feasibility of inserting the largest possible detour unit, carefully ensuring that spatial constraints are not violated and low-priority areas remain untouched. By prioritizing larger detour units when sufficient space is available, the algorithm reduces the creation of short segments, thereby preserving a wider design space for assigning different waveguide types during the subsequent automatic insertion stage. Then in Steps 2 and 3, a detour unit with a width of four grids is inserted at the candidate point. Areas that fall within this detour unit but are not directly traversed by the detour path are subsequently marked as low-priority, illustrated by the shaded red regions. After insertion, the endpoint of the newly placed detour segment automatically becomes the next candidate point, allowing a continuous and logical path extension toward the target. By incrementally inserting detour segments along the routing path, the path naturally follows the outer boundary of the layout. This ensures that each segment travels the maximum possible distance before a bend is introduced, which in turn helps minimize the total number of bends and thus reduces optical loss. Notably, in Step 4, a potential downward detour is deliberately skipped despite the availability of space, as it lies within a previously marked low-priority region. After all primary candidates have been processed, the algorithm proceeds to revisit these low-priority areas. In Steps 5 and 6, the detouring resumes from the source node again, but given the reduced available space, the algorithm selects smaller detour units, such as those two grids in width, to achieve the remaining length extension while maintaining routing feasibility.

Routing in Constrained Regions: When the routing process enters highly constrained areas with limited routing capacity, particularly those with only narrow corridors between obstacles, the standard detour method introduced before may no longer be sufficient. When the path encounters the normal neck regions (yellow regions in Fig. 4(b)), the algorithm will evaluate their spatial extent. If the region connects to large enough unexplored detour areas, the algorithm will adopt smaller units to maneuver through these narrow corridors. Note that the dead region will not be added to these areas. In extremely constrained cases, as illustrated in Fig. 6, the net encounters the neck regions with only one routing capacity, and detour units cannot be inserted due to the minimum required width of two grid cells. In such scenarios, the algorithm then activates an extended detour mode and allows the path to loop around the bounding boxes of nearby obstacles to bypass the narrow region, effectively increasing spatial flexibility and avoiding local deadlocks.

2) Routing for Matching Net Groups

Accurate optical path length matching constraint is essential in any scenario where coherent interference is utilized, such as Mach–Zehnder

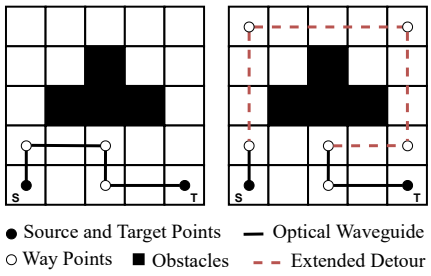

Fig. 6 Illustration of extended detour mode.

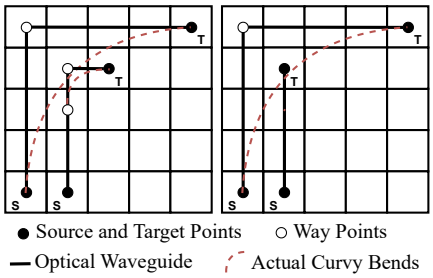

Fig. 7 Illustration of waveguide interaction near terminals.

interferometers, as even tiny mismatches distort interference and impair performance. To meet this constraint, we adopt the foundational routing framework proposed by [15] with modifications tailored to accommodate the requirements of automatic transition insertion.

In the initial phase, an A*-based global routing is applied to generate preliminary routing paths. To support the subsequent automatic transition insertion, we incorporate a piecewise segment length penalty p_s into the modified A* routing algorithm. The cost function of the A* can be computed as:

$$f(x) = c(x) + g(x') + h(x), \tag{2}$$

where x' represents the previous node of x, $g(x)$ represents the current insertion loss of the path determined by Equation (1), and $c(x)$ is the cost of the current node, which can be computed as:

$$c(x) = (1 + p_s(l(x))) \times (g(x) - g(x')). \tag{3}$$

where l is the length of the current segment, measured from the current point x back to the former bend point; $p_s(l)$ is the piece-wise constant penalty driven by the minimum-bend-radius thresholds:

$$p_s(l) = \begin{cases} P_1, & 0 < l \le R_1, \\ P_2, & R_1 < l \le R_2, \\ \vdots & \\ 0, & l > R_{k-1}. \end{cases} \tag{4}$$

Shorter segments incur the highest constant penalty P_1 to preserve downstream transition flexibility, while longer segments receive progressively smaller penalties $(P_2, \ldots, P_{k-1})$ or none at all, reflecting their greater compatibility with diverse waveguide classes. Penalties remain constant within each interval, as finer granularity does not influence automatic transition insertion.

Additionally, a terminal proximity penalty P_t is introduced to address potential crossing concerns near source and destination terminals. Fig. 7 illustrates the necessity of this modification. When two bends from distinct nets are placed in adjacent grid cells, their tightly confined curvature suppresses overlap and remains design-rule compliant. Conversely, if the straight segment that connects a source or sink runs parallel and too close to a bend of another net, unintended waveguide crossings may occur. Therefore, Equation (5) is modified as follows:

$$c(x) = (1 + p_t(x) + p_s(l(x))) \times (g(x) - g(x')). \tag{5}$$

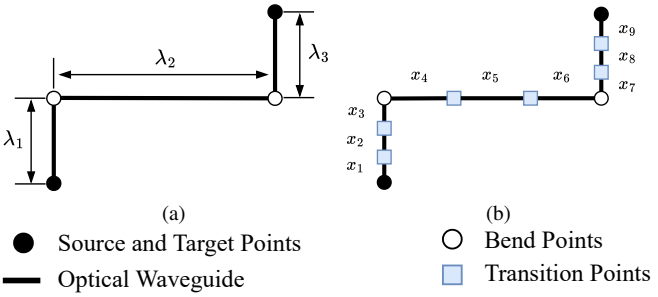

Fig. 8 Illustration of (a) segments and (b) subsegments.

where p_t is defined as:

$$p_t(x) = \begin{cases} P_{\text{term}}, & \text{if } x \in N_t, \\ 0, & \text{otherwise.} \end{cases} \tag{6}$$

where N_t is the set of neighborhoods of the source and destination terminals, and P_{term} is a constant value carefully calibrated to exceed the loss associated with a single bend but remain below that of a waveguide crossing. This encourages minor detours near terminals, effectively mitigating unintended waveguide crossings or physical design conflicts without compromising performance or introducing excessive loss.

D. Automatic Transition Insertion

After obtaining the initial topology, the next stage is to employ different waveguide types to optimize insertion loss. Although this offers flexibility, each transition between different waveguides introduces excess loss and phase shifts due to mismatches. Therefore, we formulate this task as an MILP problem, aiming to minimize total insertion loss while ensuring the matching of insertion losses across the matching group. By combining the optimization and matching tasks in our single-stage MILP framework, we can obtain the optimal solution.

The notations used in the MILP formulation are as follows:

- N: set of nets, $N = \{n_p | 1 \le p \le |N|\}$.
- S_p: set of segments in net n_p, $S_p = \{s_{p,i} | 1 \le i \le |S_p|\}$.
- T: set of waveguide types, $T = \{t_j | 1 \le j \le |T|\}$.
- G: set of nets within the loss matching group, $M = \{m_u | 1 \le u \le |M|\}$.
- $x^{pre}_{p,i,j}, x^{cur}_{p,i,j}, x^{post}_{p,i,j}$: 0-1 integer variable denoting whether waveguide type j is selected for the preceding, current, and following subsegments of the i^{th} segment in net n_p, respectively.
- $y^{pre}_{p,i}, y^{post}_{p,i}$: continuous variables denoting the normalized lengths of the preceding and following subsegments of the i^{th} segment in net n_p, respectively.
- $\lambda_{p,i}$: length of the i^{th} segment in net n_p.
- $\gamma_{j,k}$: length of the transition from waveguide type j to k.
- r_j: bend radius of waveguide type j.
- β_j: bend loss of waveguide type j.
- ϕ_j: propagation loss of waveguide type j.
- $\tau_{j,k}$: transition loss from waveguide type j to k.
- $O^{bend}_p, O^{prop}_p, O^{tran}_p, O^{total}_p$: bend loss, propagation loss, transition loss, and total loss of net n_p.

1) Subsegment Construction

We first introduce a key rule: transitions cannot be inserted to change the waveguide type during the bending process, as this would alter the topology of the bend, thereby affecting the bend loss.

Building on this rule, we model each net as a set of interconnected segments, as shown in Fig. 8(a). The segment is divided according to the source and target points as well as bend points, and the length of the i^{th} segment is described as λ_i.

For each segment, we then construct three subsegments, as illustrated in Fig. 8(b). A segment consists of two transition points and three subsegments. Specifically, we define three variables to describe the type of subsegments, including the preceding subsegment $x^{pre}_{p,i,j}$, the current subsegment $x^{cur}_{p,i,j}$, and the following subsegment $x^{post}_{p,i,j}$.

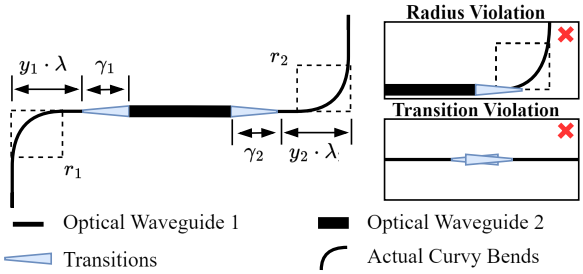

Fig. 9 Illustration of transition movement and related rules.

Since the waveguide type must remain unchanged during the bending process (e.g., $x_3 = x_4$ and $x_6 = x_7$), the following constraint should be satisfied:

$$x^{post}_{p,i,j} = x^{pre}_{p,i+1,j}, \quad i \in \{1, \dots, |S_p| - 1\}, \tag{7}$$

where p represents the p^{th} net in N, i is the i^{th} segment in S_p, and j refers to the j^{th} waveguide type in T. The condition $i \in \{1, \dots, |S_p| - 1\}$ ensures that the constraint applies to each bend.

Moreover, only one type of waveguide is allowed per subsegment. Therefore, the following constraints must be held for each subsegment:

$$\sum_{j=1}^{|T|} x^{\alpha}_{p,i,j} = 1, \tag{8}$$

where $\forall \alpha \in \{pre, cur, post\}$, and $i \in \{1, \dots, |S_p|\}$.

2) Transition Movement

The discrete nature of loss changes resulting from waveguide type selection makes satisfying the loss matching constraint difficult. Considering this, we model the position of transitions using continuous variables.

As shown in Fig. 9, the segment consists of 3 subsegments and 2 transitions, while the waveguide type of the subsegments changes from type 1 to type 2 and then back to type 1 from left to right. Moreover, $y_1 \cdot \lambda$ and $y_2 \cdot \lambda$ represent the length of the preceding subsegment and following subsegment, respectively. The radii of bends are presented as r_1 and r_2. The lengths of transitions are presented as γ_1 and γ_2.

For the variables y_1 and y_2, they are normalized to the range $[0, 1]$ using the length of the corresponding segment λ_1, where 0 indicates the transition is at the proximal bend, and 1 indicates the transition is at the distal bend.

However, if y_1 or y_2 is 0 or 1, the transition will be shifted to the bend, which is prohibited and called "Radius Violation" as illustrated in Fig. 9. Therefore, the constraints below should be followed:

$$R^{\alpha}_{p,i} = \sum_{j=1}^{|T|} (x^{\alpha}_{p,i,j} \cdot \frac{r_j}{\lambda_{p,i}}), \tag{9}$$

$$y^{\alpha}_{p,i} \ge R^{\alpha}_{p,i}, \quad \forall \alpha \in \{pre, post\}, \tag{10}$$

$$y^{pre}_{p,i} \le 1 - R^{post}_{p,i}, y^{post}_{p,i} \le 1 - R^{pre}_{p,i}, \tag{11}$$

where Equation (9) represents the bend radii of the waveguides corresponding to the previous and following subsegments, respectively, and the bend radii r are normalized using λ. Furthermore, as indicated by Equation (8), the sum of x for each subsegment must equal 1, meaning that the value of R is equal to the normalized radius of the corresponding waveguide type. Finally, Equation (10) and Equation (11) ensure that the position of the preceding transition $y^{pre}_{p,i}$ and the following transition $y^{post}_{p,i}$ lies between the proximal and distal bends.

Additionally, the "Transition Violation" illustrated in Fig. 9 should also be considered, which is the situation that the length of the current subsegment cannot accommodate the corresponding transitions. It can

be defined as follows:

$$C_{p,i}^{\alpha} = \sum_{j=1}^{|T|} \sum_{\substack{k=1 \\ k \neq j}}^{|T|} (x_{p,i,j}^{cur} \cdot x_{p,i,k}^{\alpha} \cdot \frac{\gamma_{j,k}}{\lambda_{p,i}}), \forall \alpha \in \{pre, post\}, \tag{12}$$

$$1 - y_{p,i}^{pre} - y_{p,i}^{post} \geq C_{p,i}^{pre} + C_{p,i}^{post}, \tag{13}$$

where Equation (12) represents the normalized transition lengths of the preceding transition and following transition, respectively. The product of x is used to detect the transition type. Finally, Equation (13) indicates that the length of the current subsegment must be greater than or equal to the sum of the transition lengths.

3) MILP-based Automatic Transition Insertion

After defining the waveguide type and transition movement, we define the objective function to minimize the total insertion loss O_p^{total}, which includes bending loss O_p^{bend}, propagation loss O_p^{prop}, and transition loss O_p^{tran}. Crossed paths are divided into separate paths, and crossing points are treated as new start or end points, so the crossing loss is excluded. The total loss O_p^{total} is defined as follows:

$$O_p^{total} = O_p^{bend} + O_p^{prop} + O_p^{tran}. \tag{14}$$

The bending loss O_p^{bend} is defined as follows:

$$O_p^{bend} = \sum_{i=2}^{|S_p|} \sum_{j=1}^{|T|} \left(x_{p,i,j}^{pre} \cdot \beta_j \right), \tag{15}$$

where β_j is the bend loss of j^{th} waveguide type.

The propagation loss O_p^{prop} is defined as follows:

$$h_{p,i}^{\alpha} = y_{p,i}^{\alpha} - R_{p,i}^{\alpha}, \quad \forall \alpha \in \{pre, post\}, \tag{16}$$

$$h_{p,i}^{cur} = 1 - y_{p,i}^{pre} - y_{p,i}^{post} - C_{p,i}^{pre} - C_{p,i}^{post}, \tag{17}$$

$$V_{p,i}^{\alpha} = \sum_{j=1}^{|T|} \left(x_{p,i,j}^{\alpha} \cdot \phi_j \cdot h_{p,i}^{\alpha} \cdot \lambda_{p,i} \right), \forall \alpha \in \{pre, cur, post\}, \tag{18}$$

$$O_p^{prop} = \sum_{i=1}^{|S_p|} \left(V_{p,i}^{pre} + V_{p,i}^{cur} + V_{p,i}^{post} \right), \tag{19}$$

where $h_{p,i}^{pre}$ and $h_{p,i}^{post}$ in Equation (16) refers to the normalized length of the preceding and following subsegment, after excluding the radii of the bends $R_{p,i}^{pre}$ and $R_{p,i}^{post}$, respectively. $h_{p,i}^{cur}$ in Equation (17) refers to the normalized length of the current subsegment, after excluding the lengths of the transitions $C_{p,i}^{pre}$ and $C_{p,i}^{post}$. For Equation (18), the product of x and ϕ represents the propagation loss of the corresponding waveguide type, and the product of h and λ represents the actual length of the subsegment. Following this, we can obtain the propagation loss of the p^{th} net according to Equation (19).

The transition loss O_p^{tran} is defined as follows:

$$O_p^{tran} = \sum_{i=1}^{|S_p|} \sum_{j=1}^{|T|} \sum_{\substack{k=1 \\ k \neq j}}^{|T|} \left(\tau_{j,k} \cdot x_{p,i,j}^{cur} \cdot \left(x_{p,i,k}^{pre} + x_{p,i,k}^{post} \right) \right), \tag{20}$$

where the product of x can detect the waveguide type transition, and multiplying by $\tau_{j,k}$ can give the loss for each transition.

Finally, to ensure that the insertion losses are matched across the matching group G, the loss matching constraint is as follows:

$$|O_p^{total} - O_q^{total}| \leq \epsilon, \quad n_p, n_q \in G, \tag{21}$$

where ϵ is a sufficiently small positive number.

4) Problem Linearization

The formulation of all constraints is provided in the previous sections. Consequently, the final optimization model is given by:

$$\begin{aligned} &\text{minimize:} && \sum_{p=1}^{|N|} O_p^{\text{total}}, \\ &\text{subject to:} && (7)\text{–}(13), (21). \end{aligned} \tag{22}$$

To handle the non-linear variables in the formulation, we apply a linearization approach by introducing auxiliary variables and utilizing the "big M method" [20], thus transforming the non-linear terms into linear ones.

E. Postprocessing

In the post-processing phase, we determine the actual physical connections guided by the final result from the MILP solver. Given that there is only one line in each edge per grid, the process is straightforward. We convert the grid-based coordinates to actual physical coordinates and adjust the coordinates of turning points accordingly. For turning points on the boundary of the path, slight adjustments are made to ensure length-matching constraints are maintained.

IV. Experimental Results

The initial routing component was implemented in C++ and executed on a Linux server equipped with a 2.00 GHz Intel Xeon Gold 6338 CPU and 256 GB of RAM. The remaining components were implemented in Python and executed on a computer with an Intel Core i7-13700 CPU and 32 GB of RAM. The MILP problem was solved using Gurobi [21].

There are three waveguide types with different etch depths (full, deep, and shallow) in the PDK of an industrial foundry. These waveguides exhibit bend losses of 0.01, 0.02, and 0.04 dB, propagation losses of 1.67, 1.02, and 0.57 dB/cm, and bend radii of 10, 15, and 40 μm, respectively. Crossings are allowed only in full-etch lines, and the crossing loss is 0.2 dB. Transition losses are 0.01 dB (full-deep), 0.04 dB (full-shallow), and 0.05 dB (deep-shallow). The latter transition uses a 40×11 micrometers specification, while others are 20×11 micrometers.

A. Single-Net Target-Length Routing Results

To evaluate the effectiveness of our target-length routing algorithm, we compare it directly against the CAFE router [18] on two key metrics: total optical loss and computation time, both before and after automatic transition insertion (ATI). Four representative testcases were generated on grid layouts (100×40 and 50×50) populated with randomly placed rectangular obstacles, with two scenarios varying only the target path length (PIC 1–2) and two varying both obstacle distribution and length (PIC 3–4). In each case, the target waveguide length was fixed in advance, and the obstacle density was chosen to reflect realistic photonic-circuit constraints.

TABLE I summarizes the comparison results and the testcases. As shown in the table, even without the ATI, our proposed method consistently achieves significantly lower optical loss by approximately 14% compared to the CAFE router, while also reducing runtime from several hundred seconds to under 3 seconds per routing. This substantial speedup and loss reduction highlight the intrinsic quality of our routing strategy, which effectively reduces bend counts during routing.

When ATI is applied, both routing methods exhibit further improvements in optical loss, highlighting its effectiveness in improving overall routing performance. Specifically, ATI yields an average reduction of 21% for the CAFE router and 23% for our method, relative to their respective pre-ATI results. Notably, while both benefit from the optimization, the absolute reduction achieved by our approach is greater, despite already starting from a lower initial loss. This further improvement highlights the compatibility of our routing strategy with the post-routing ATI. Fig. 10 shows our routing results of PIC1 and PIC4. The blue, green, and red lines represent full-, deep-, and shallow-etched waveguides, respectively, while the black rectangles indicate transitions between different waveguides.

B. Results of Multi-Nets Routing with Matching Constraints

To evaluate the effectiveness of our method for nets with matching constraints in PICs, we collected six representative real-world PIC

TABLE I Comparison of target length routing metrics between CAFE Router [18] and our method.

Cases	Grid Size ($w \times h$)	# Obstacle Grids	Target Length (μm)	CAFE Router w/o ATI^*		CAFE Router w/ ATI		Ours w/o ATI		Ours w/ ATI	
				Loss (dB)	Runtime (s)	Loss (dB)	Runtime (s)	Loss (dB)	Runtime (s)	Loss (dB)	Runtime (s)
PIC1	100×40	1048	24000	5.05	480.5	4.02	480.7	4.53	0.6	**3.29**	1.1
PIC2	100×40	1048	57600	12.38	707.4	10.08	708.6	11.35	2.1	**8.96**	3.6
PIC3	50×50	878	30800	7.80	288.9	7.08	298.5	6.52	0.8	**5.75**	1.4
PIC4	50×50	758	32000	6.96	538.4	6.01	539.0	6.55	1.0	**5.50**	1.7
Norm.				1.37	258.4	1.16	259.8	1.23	0.6	1.0	1.0

*ATI represents the automatic transition insertion.

TABLE II Comparison of the total and max insertion loss (TL, ML), loss mismatch (L_{mis}), and runtime for different methods.

Cases	$\#nets$	Router in [15]				DP + Refinement				Ours			
		TL (dB)	ML (dB)	L_{mis} (%)	Runtime (s)	TL (dB)	ML (dB)	L_{mis} (%)	Runtime (s)	TL (dB)	ML (dB)	L_{mis} (%)	Runtime (s)
PIC5	5	1.64	**0.27**	**0**	1.429	**1.57**	**0.27**	3.25	1.430	1.63	**0.27**	**0**	1.615
PIC6	8	2.48	0.31	**0**	2.011	**2.12**	**0.26**	**0**	2.012	**2.12**	**0.26**	**0**	2.127
PIC7	8	8.26	1.03	**0**	29.156	**5.66**	**0.71**	**0**	29.157	**5.66**	**0.71**	**0**	29.612
PIC8	8	18.14	1.14	0.71	20.549	**16.31**	**1.10**	7.40	19.619	17.61	**1.10**	**0**	20.780
PIC9	16	11.52	0.77	7.02	4.471	**13.34**	**0.75**	13.73	3.619	11.99	**0.75**	**0**	4.872
PIC10	16	16.39	2.37	13.56	1.023	**15.45**	**2.05**	5.91	1.029	16.41	**2.05**	**0**	7.339
Norm.		1.05	1.14	1.0	0.88	0.98	1.0	1.27	0.86	1.0	1.0	0	1.0

*L_{mis} represents the average ratio of the loss differences between nets to the maximum loss among all nets.

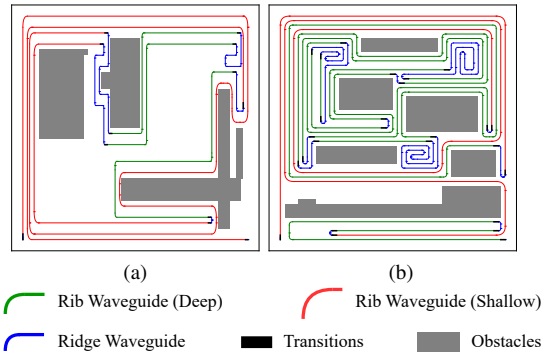

Fig. 10 Results for (a) PIC1 and (b) PIC4.

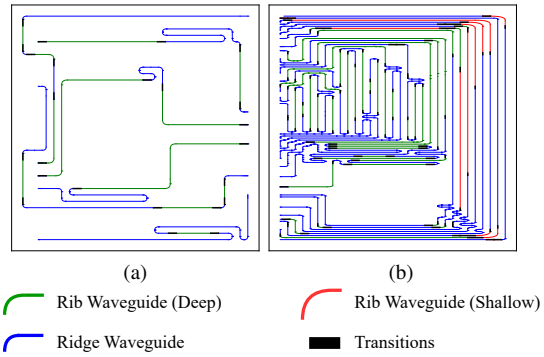

Fig. 11 Results for (a) PIC5 and (b) PIC9.

designs, each containing between 5 and 16 nets requiring matching, with varying levels of structural complexity. The performance of layouts generated by our adaptive waveguide routing framework is compared against two baselines, including the method proposed in [15]. Although the method in [15] is not specifically designed for insertion loss matching, its focus on bend and length matching makes it applicable for achieving comparable results. Under fixed path-length constraints and without ATI, bend matching becomes the primary mechanism for controlling insertion loss, as waveguide crossings simultaneously affect both intersecting nets. In addition to [15], we implement a two-stage "DP + Refinement" strategy for a more comprehensive comparison. This approach first assigns waveguide types with dynamic programming to minimize insertion loss across all nets, then shifts transitions to raise lower-loss paths and meet loss-matching constraints. Both baseline strategies, as well as our proposed method, are applied to the same initial routing results to ensure a fair and consistent basis for comparison.

TABLE II shows the comparison between routing results. As shown in the table, our method minimizes total insertion loss while strictly satisfying the loss matching constraint in all 6 cases. In contrast to other approaches, our method achieves zero average loss mismatch across all test cases, demonstrating perfect uniformity in insertion loss among nets. Moreover, our method yields a 5% reduction in total loss and a 14% reduction in maximum loss compared to the baseline in [15], indicating its efficiency. The two-stage "DP + Refinement" method achieves moderate loss reduction but fails to consistently satisfy the loss matching constraint. This limitation arises because the method focuses on minimizing individual path losses in isolation, without jointly considering the trade-off between loss minimization and interconnect uniformity. The inability of the DP + refinement method to satisfy all matching requirements further highlights the advantage of our integrated MILP formulation, which captures the joint optimization space of waveguide selection and constraint enforcement more effectively. Fig. 11 shows our routing results of PIC5 and PIC9. Regardless of how the pins of the matching net groups are arranged, our methods can generate high-quality routing results with minimized insertion loss while adhering to matching constraints.

V. Conclusion

This paper presents a complete optical routing framework tailored for PICs with hybrid waveguide classes. The proposed initial routing strategies effectively minimize bends and short segments while meeting strict target lengths or length matching constraints. Our MILP-based automatic transition insertion further refines the routing by optimally assigning waveguide types, significantly reducing overall insertion loss while ensuring compliance with stringent loss matching constraints. Comprehensive experiments validate the effectiveness of our framework across diverse testcases, showing that compared to prior approaches, our algorithm reduces total insertion loss by up to 37% on average in target-length routing scenarios, and achieves an average reduction of 5% in total insertion loss and 14% in maximum loss in multi-net cases while satisfying all constraints.

Acknowledgment

This work is supported in part by the Nansha District Key Area S&T Scheme (No. 2024ZD007) and the Guangdong Science and Technology Department (No.2025B1212150003).

REFERENCES

[1] L. Thylén, S. He, L. Wosinski, and D. Dai, "The moore's law for photonic integrated circuits," *Journal of Zhejiang University-SCIENCE A*, 2006.

[2] P. L. McMahon, "The physics of optical computing," *Nature Reviews Physics*, 2023.

[3] K. Lu, Z. Chen, H. Chen, W. Zhou, Z. Zhang, H. K. Tsang, and Y. Tong, "Empowering high-dimensional optical fiber communications with integrated photonic processors," *Nature Communications*, 2024.

[4] Z. Xu, T. Zhou, M. Ma, C. Deng, Q. Dai, and L. Fang, "Large-scale photonic chiplet taichi empowers 160-tops/w artificial general intelligence," *Science*, 2024.

[5] P. Team, "A manufacturable platform for photonic quantum computing," *Nature*, 2025.

[6] S. K. Selvaraja and P. Sethi, "Review on optical waveguides," *Emerging Waveguide Technology*, 2018.

[7] W. Bogaerts and L. Chrostowski, "Silicon photonics circuit design: methods, tools and challenges," *Laser & Photonics Reviews*, 2018.

[8] K. Zetie, S. Adams, and R. Tocknell, "How does a mach-zehnder interferometer work?" *Physics Education*, 2000.

[9] D. H. Geuzebroek and A. Driessen, "Ring-resonator-based wavelength filters," in *Wavelength filters in fibre optics*. Springer, 2006, pp. 341–379.

[10] Y. Yi, D. Wu, V. Kakdarvishi, B. Yu, Y. Zhuang, and A. Khalilian, "Photonic integrated circuits for an optical phased array," *Photonics*, 2024.

[11] P. R. Lawson, "Phase and group delay estimation," *Principles of long baseline stellar interferometry*, 2000.

[12] Y. Wu, X. Yu, H. Chen, Y. Luo, Y. Tong, and Y. Ma, "PICBench: Benchmarking LLMs for photonic integrated circuits design," *2025 Design, Automation & Test in Europe Conference (DATE)*, 2025.

[13] 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 *Proceedings of the 46th annual design automation conference*, 2009.

[14] F.-Y. Chuang and Y.-W. Chang, "On-chip optical routing with waveguide matching constraints," in *2021 IEEE/ACM International Conference On Computer Aided Design (ICCAD)*, 2021.

[15] Y. Wu, W. Guan, Y. Tong, and Y. Ma, "Automatic routing for photonic integrated circuits under delay matching constraints," in *Design, Automation and Test in Europe Conference and Exhibition*, 2025.

[16] C. Du, Y. Cai, and X. Hong, "A novel analog routing algorithm with constraints of variable wire widths," in *2006 International Conference on Communications, Circuits and Systems*, vol. 4, 2006.

[17] Q. Liu, D. Lin, C. Chen, H. He, J. Chen, and Y.-W. Chang, "A matching based escape routing algorithm with variable design rules and constraints," in *2023 60th ACM/IEEE Design Automation Conference (DAC)*, 2023.

[18] Y. Kohira, S. Suehiro, and A. Takahashi, "A fast longer path algorithm for routing grid with obstacles using biconnectivity based length upper bound," in *Proceedings of the 2009 Asia and South Pacific Design Automation Conference*, ser. ASP-DAC '09. IEEE Press, 2009.

[19] C. Y. Lee, "An algorithm for path connections and its applications," *IRE transactions on electronic computers*, 1961.

[20] I. Griva, S. G. Nash, and A. Sofer, *Linear and nonlinear optimization 2nd edition*. SIAM, 2008.

[21] Gurobi Optimization, LLC, "Gurobi Optimizer Reference Manual," 2024. [Online]. Available: https://www.gurobi.com