Concurrent Detailed Routing with Pin Pattern Re-generation for Ultimate Pin Access Optimization

Ying-Jie Jiang
D10907011@mail.ntust.edu.tw
Department of Electrical Engineering, National Taiwan
University of Science and Technology, Taipei, Taiwan

Shao-Yun Fang
syfang@mail.ntust.edu.tw
Department of Electrical Engineering, National Taiwan
University of Science and Technology, Taipei, Taiwan

ABSTRACT

Pin access has become one of the most significant challenges in large-scale full-chip routing due to the continuous reduction in feature sizes and the increasing complexity of designs. The conventional standard cell layout synthesis approaches usually optimize pin accessibility by maximizing pin lengths and access points. However, these pre-determined pin patterns greatly occupy routing resources and may contrarily degrade routability. To address this problem, this paper proposes the first work of concurrent detailed routing with pin pattern re-generation to achieve ultimate pin access optimization. A pseudo-pin extraction and routing technique is proposed that can secure one access point for each input/output pin while allowing the remaining access points to be routable by other nets. The experimental results demonstrate that the proposed method can resolve 89% of local regions that are unroutable with original layout patterns without compromising power and timing performances.

1 INTRODUCTION

In recent decades, there has been remarkable progress in semiconductor technology, leading to the miniaturization of electronic devices. This advancement has been primarily driven by continuous improvements in transistor and interconnect dimensions, resulting in highly compact integrated circuits. While numerous benefits have been brought such as increased functionality and improved performance, significant challenges in circuit design are also present. In the cell-based design flow, standard cells play a crucial role as fundamental building blocks in digital integrated circuits. However, as technology continues to scale down, the greatly decreased physical dimensions of standard cells result in a significant increase in pin density, which poses considerable difficulty in addressing the pin access problem during routing. Existing approaches to optimize pin accessibility can be roughly categorized into two strategies: (1) cleverly designing pin patterns of standard cells, and (2) conducting pin access-aware routing.

The existing studies using the first strategy develop pin access-driven layout synthesis techniques for standard cells. Since each pin can be accessed from any direction, it is exceptionally challenging for these approaches to determine the most appropriate pin pattern for detailed routing. Consequently, they often rely on techniques that yield longer pin patterns. For example, [7] sets the constraint on the minimum I/O pin length and [17] maximizes pin lengths by using a linear programming (LP) formulation. Similarly, [12] increases the number of intersections of each pin with Metal-2 tracks, and [10] ensures minimal access points via a Boolean satisfiability (SAT) formulation. In addition, [2] establishes a measure called minimum I/O pin opening to ensure that every access point can be reached without violating design rules. [11] strategically introduces gaps or spaces within a single cell can alleviate potential congestion and ensure better access to pins. However, pin pattern optimization in the standard cell layout synthesis stage suffers from two major drawbacks. First, although the intentions made by the above studies could enhance overall pin accessibility with larger numbers of valid access points, they still cannot guarantee pin access due to unpredictable router behavior. Second, longer pin patterns actually consume more routing resources on lower metal layers, which may contrarily lower the overall routability.

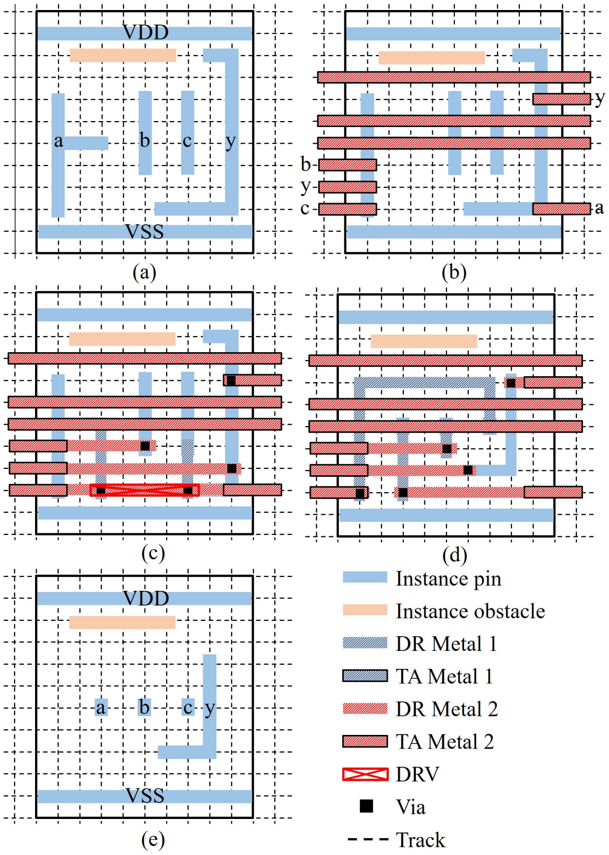

Figure 1: An example of concurrent detailed routing (DR) with pin pattern re-generation. (a) An instance with four pins *a*, *b*, *c*, and *y* from left to right. (b) An instance after track assignment (TA). (c) A detailed routing result after the original pin pattern. (d) A detailed routing result after the proposed flow. (e) The re-generated pin pattern after the proposed flow.

The existing approaches using the second strategy either pre-determine design rule violation (DRV)-free access points for the pins of a row/group of cells simultaneously before detailed routing, or directly consider pin accessibility during detailed routing. To pre-determine DRV-free access points, [6] introduces a dynamic programming-based method by considering boundary conflicts, and [13] develops an SAT formulation. In addition to pre-determining pin access points, a number of works further connect the access points with Metal-2 segments to maximize pin access flexibility [1, 9, 14, 16]. However, fixing DRV-free access

DAC '24, June 23–27, 2024, San Francisco, CA, USA

ACM ISBN 979-8-4007-0601-1/24/06…$15.00
https://doi.org/10.1145/3649329.3655918

Table 1: The notations of the ILP formulation.

Term	Description
$G(V, E)$	The multi-layer routing graph with a vertex set V and an edge set E
v_i	The i-th vertex in V
$e_{i,j}$	The edge between v_i and v_j in E
$fe_{i,j}$	0-1 variable to determine if the physical metal edge $e_{i,j}$ is used
$c_{i,j}$	Cost of $fe_{i,j}$
C	The set of connections
$G^c(V^c, E^c)$	The multi-layer routing graph of a connection c with with a vertex set V_c and an edge set E_c
O^c	The set of obstacle vertices for connection c
v_i^c	The i-th vertex in V^c
$e_{i,j}^c$	The edge between v_i^c and v_j^c in E^c
fv_i^c	0-1 variable to determine if vertex v_i^c is used by connection c
$fe_{i,j}^c$	0-1 variable to determine if edge $e_{i,j}^c$ is used by connection c
SV^c	The set of super vertices of connection c
N_i^c	The set of neighboring vertices of v_i^c
DNC	The set of different-net connection combinations
NC	A different-net connection combination

points before routing strictly limits the solution space, and extending access points with Metal-2 segments becomes another waste of routing resources since they can hardly be fully utilized during detailed routing. To address pin accessibility during the detailed routing process, [15] introduces a pin-access graph (PAG) and a pin-access-driven rip-up and reroute scheme. This dynamic approach determines pin accessibility in real-time during detailed routing. Moreover, [5] proposes an integer linear programming (ILP)-based detailed router that concurrently addresses spatially related nets to guarantee 100% pin accessibility if a solution exists.

Given that each of the two existing strategies exhibits its advantages and disadvantages, it would be exciting if the strengths of both strategies could be exploited to simultaneously maximize pin accessibility and optimize routing performance. Figure 1 illustrates a comprehensive example motivating this work. Figure 1(a) shows a standard cell with four pins a, b, c, and y. Figure 1(b) gives a detailed routing (DR) instance after track assignment (TA), where the short segments need to be routed to the four pins, and the long segments are other nets passing this cell. It can be observed from Figure 1(c) that a DRV-free solution cannot be found by applying conventional detailed routing. However, if the routing resource occupied by the original Metal-1 pins can be released, a valid routing solution in Figure 1(d) does exit, which is achieved as if the pin pattern is re-generated as that shown in Figure 1(e).

The instance in Figure 1 suggests an interesting insight that has never been considered: in some cases, it is almost impossible to find a routing solution if the pin pattern of a cell is pre-determined. Therefore, for a local hard-to-access region, where no solution is found after numerous rounds of sequential routing or after optimal concurrent routing, we may improve the conventional design flow by performing detailed routing prior to fixing the pin pattern of the region. The major contributions of this paper are listed as follows:

(1) This paper presents the first work in the literature on concurrent detailed routing and pin pattern re-generation to maximize the overall pin accessibility of a given design. The concurrent scheme can result in an optimal routing result with the maximized routing resource on the lowest layers and an automatically generated pin pattern.
(2) The proposed ILP formulation allows other connections by utilizing the resources occupied by the original pin patterns. Additionally, the regenerated pin pattern considers the placement of transistors, thereby ensuring the functionality of the standard cell.
(3) The experimental results demonstrate that the proposed approach effectively addresses 89% of local regions that were initially unrouteable by employing the state-of-the-art concurrent detailed router [5]. This achievement comes with a reasonable runtime increase of 31%.

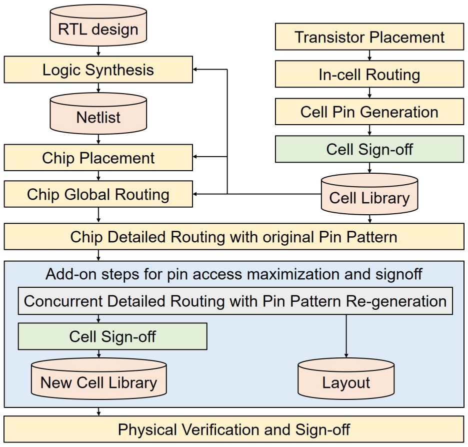
RTL design
Logic Synthesis
Netlist
Chip Placement
Chip Global Routing
Transistor Placement
In-cell Routing
Cell Pin Generation
Cell Sign-off
Cell Library
Chip Detailed Routing with original Pin Pattern
Add-on steps for pin access maximization and signoff
Concurrent Detailed Routing with Pin Pattern Re-generation
Cell Sign-off
New Cell Library
Layout
Physical Verification and Sign-off

Figure 2: The proposed concurrent detailed routing with pin pattern re-generation is applied within the conventional cell-based design flow.

(4) We also show that the electrical characteristics of standard cells do not significantly change after such a pin pattern re-generation process with an industrial simulation tool.

The subsequent sections of this paper are structured as follows: Section 2 presents the ILP formulation of the concurrent detailed router [5], based on which our approach will be constructed. Section 3 introduces an overall flow of the proposed detailed routing with pin pattern re-generation. Section 4 delves into the key techniques employed within the proposed flow. Section 5 presents the experimental results. Finally, Section 6 offers the concluding remarks.

2 PRELIMINARY

[5] has presented an ILP formulation to solve the pin access-driven concurrent detailed routing problem based on a multi-commodity flow model, which can simultaneously identify suitable access points and perform detailed routing. Compared to other concurrent routers in the literature, [5] is able to automatically connect and optimize multi-pin nets which are essential in detailed routing. Table 1 shows the notations employed in the ILP formulation, which is performed on a multi-layer routing graph $G(V, E)$. The fundamental constraints and the objective function are briefly introduced as follows:

Flow conservation constraint: For a super vertex that is the source or a target of a connection c, Eq. (1) makes the total flow of c incident to the super vertex is 1. For a basic vertex other than super vertices, Eq. (2) ensures the total flow of c incident to the basic vertex is either 0 or 2.

$$\sum_{v_a^c \in N_i^c} fe_{i,a}^c = 1, \forall v_i^c \in SV^c, \forall c \in C. \tag{1}$$

$$\sum_{v_a^c \in N_i^c} fe_{i,a}^c - 2 \times fv_i^c = 0, \forall v_i^c \in V^c, \forall c \in C. \tag{2}$$

Obstacle constraint: Eq. (3) stipulates that, for each connection c, the total flow of c incident to a basic vertex must equal 0 if the vertex is identified as an obstacle for c. Such a vertex may be a real obstacle or a routing grid belonging to other net connections, it is important to emphasize that a net connection belonging to the same net is not considered an obstacle in this context.

$$\sum_{v_a^c \in N_i^c} fe_{i,a}^c = 0, \forall v_i^c \in O^c, \forall c \in C. \tag{3}$$

Exclusive constraint: Eq. (4) and Eq. (5) forbids connections of different nets sharing vertices and edges. These two equations actually enable the multiple 2-pin connections of a single net to share vertices

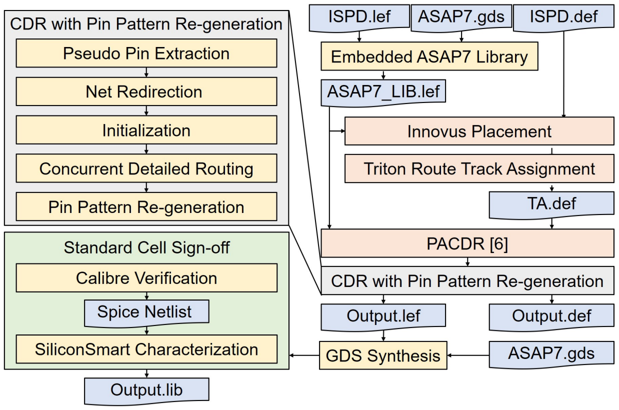

Figure 3: The overall experimental flow using the ASAP7 cell library and ISPD'18 benchmarks.

and edges such that a minimum Steiner tree can be automatically generated. (Refer to [5] for more details.)

$$\sum_{c\in NC} fe^{c}_{i,j} \le 1, \forall e_{i,j} \in E, \forall NC \in DNC. \tag{4}$$

$$\sum_{c\in NC} fv^{c}_{i} \le 1, \forall v_{i} \in V, \forall NC \in DNC. \tag{5}$$

Physical constraint: The physical constraint determines the usage of each physical metal edge. Eq. (6) forces $fe_{i,j}$ to be 1 if any connection uses the edge.

$$fe^{c}_{i,j} \le fe_{i,j}, \forall e_{i,j} \in E, \forall c \in C. \tag{6}$$

Minimization function: Finally, the objective aims to minimize the total usage of physical edges and thus minimize the total wirelength and the via count.

$$\min. \sum_{e_{i,j}\in E} c_{i,j} \times fe_{i,j}. \tag{7}$$

3 OVERALL DESIGN FLOW

Figure 2 shows how the proposed concurrent detailed routing and pin pattern re-generation technique can be added to the traditional design flow, which has been highlighted with the blue box. After conventional detailed routing with original pin patterns, the unroutable local regions are identified and re-routed with the proposed approach. After that, we re-characterized each standard cell with the re-generated pin pattern for sign-off verification.

The comprehensive experimental process is depicted in Figure 3. To implement the design flow, we utilize the ASAP7_PDK [20] as our cell library and the ispd_18 benchmark [19] as our design. The input of our concurrent detailed routing with pin pattern generation consists of an LEF file containing technology information and a DEF file describing the track assignment results. We generate the LEF file by embedding the ispd_18 LEF file (ISPD.lef) with the ASAP7 cell library GDS file (ASAP7.gds), resulting in an embedded LEF file (ASAP7_LIB.lef) with the ASPP7 cell library pin pattern. For the track assignment DEF file, we employ the commercial tool INNOVUS [22] for cell placement and the open-source detailed router TritonRoute-WXL [21]. We input the embedded LEF and DEF files of the adopted benchmarks into INNOVUS. Although the original DEF file includes placement information, overlaps can occur among cells after embedding the ASAP7 cell library. Hence, we use INNOVUS to re-place the entire design and obtain a legalized placement result, which is then fed into TritonRoute-WXL. This process generates another DEF file (TA.def) after performing track assignment with TritonRoute-WXL. Although the conventional detailed routing with the original pin patterns can be achieved by applying any detailed router, in this work, we employed the concurrent detailed router PACDR proposed in [5] to validate the effectiveness of pin pattern re-generation. Following that, our proposed concurrent detailed routing (CDR) with pin pattern re-generation resolves the previously unsolvable regions by using the Metal-1 resources that were originally occupied by the original pin patterns and then re-generating the pin patterns in these regions.

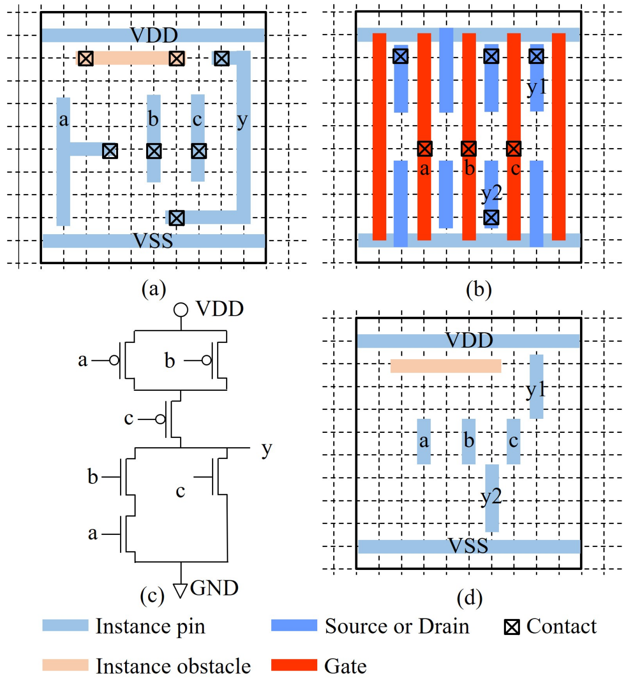

Figure 4: The layout, schematic and the pseudo pin extraction result for the cell AOI21xp5_ASAP7_75t_R in the ASAP7 cell library. (a) The in-cell routing and pin patterns on Metal-1. (b) The transistor placement on Metal-0. (c) The schematic view. (d) The extracted pseudo pins.

The concurrent detailed routing with pin pattern re-generation flow is shown on the left side of Figure 3. First, the pseudo-pins are extracted to retrieve the transistor pin information from the underlying transistor placement. In the net redirection phase, pin connections associated with a pin pattern are added to their respective nets. The initialization stage involves setting up connections, trimming redundant metals, routing trivial connections, and applying the R-tree spatial clustering technique described in [5]. Subsequently, concurrent detailed routing using the maximized routing resource released from the original pin patterns is conducted for each pin cluster, and the re-generated pin layout can be automatically determined at the same time. The resulting LEF file (Output.lef) describes the re-generated pin patterns, whereas the ASAP7 cell library GDS file (ASAP7.gds) contains the original transistor placements. By synthesizing these two files together, a multitude of unique cells are generated. To validate these cells, we utilize the commercial verification tool Calibre [23] to conduct design rule check (DRC), layout versus schematic (LVS), and parasitic extraction (PEX). Moreover, we utilize the commercial characterization tool SiliconSmart [24] and the simulation tool HSPICE [25], incorporating BSIM-CMG [18] as the advanced FinFET physical effect model. This comprehensive approach is utilized for the re-characterization of standard cells with re-generated pin patterns.

4 CONCURRENT DETAILED ROUTING WITH PIN PATTERN RE-GENERATION

The critical steps in the proposed concurrent detailed routing with pin pattern re-generation flow are introduced in the following subsections.

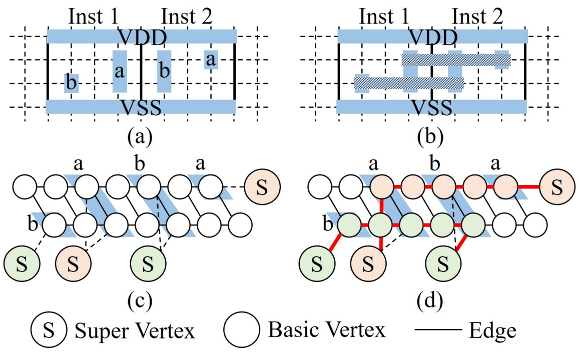

Figure 5: The flexibility of pseudo-pin patterns for routability optimization. (a) Two cell instances with two nets *a* and *b* to be connected using only Metal-1. (b) The connection results derived from Figure 5(d). (c) A multi-commodity flow model for Figure 5(a). (d) Because of the flexibility of pseudo pin patterns, Net *a* can be routed over Pin *b*, and vice versa.

4.1 Pseudo Pin Extraction

To maximally utilize the routing resource occupied by the original pin patterns while ensuring cell connectivity, we extract pseudo-pins for each cell according to its beneath transistor layout. Figure 4 gives an example of the AOI21xp5_ASAP7_75t_R cell in the ASAP7 library, where the original pin patterns and in-cell routing, the transistor placement, and the schematic are separately presented in Figures 4(a)–(c). To facilitate the following explanation, the connections for in-cell routing in combination with pin patterns are classified into four types as follows:

(1) **Type 1 An in-cell routing and a pin pattern are both required:** This type requires a pin pattern that also routes an in-cell net to connect transistor diffusions. Pin y in Figure 4(a) belongs to this type, which connects the diffusions $y1$ and $y2$ in Figure 4(b) and becomes the output pin simultaneously.

(2) **Type 2 Only an in-cell routing is required:** This type routes an in-cell net, but no pin pattern is required. The orange metal segment in Figure 4(a) belongs to this type, which serves as an obstacle during routing and pin pattern re-generation.

(3) **Type 3 Only a pin pattern is required:** This type requires a pin pattern but not an in-cell routing. Pins a, b, and c in Figure 4(a) belong to this type.

(4) **Type 4 No need for both in-cell routing and pin pattern:** This type achieves an in-cell connection during the transistor placement phase, and no additional operation is required afterward. The connection between the nMOS a and b in Figure 4(c) belongs to this type.

Our primary goal of pin pattern re-generation is to generate the minimum essential patterns to establish connections. Obviously, we do not need to consider Type-4 connections. For each Type-2 connection, although it is possible to be rerouted in our concurrent detailed routing flow, it may degrade cell characteristics because it has usually been optimized in the original cell layout. Consequently, we choose to fix Type-2 connections in our design flow. Therefore, we focus on extracting the pseudo-pins in Type-1 and Type-3. To achieve this, the corresponding location of the associated diffusion or gate of each input/output pin is found and is treated as the pseudo-pin. Figure 4(d) illustrates the extracted pseudo-pin patterns of the AOI21xp5_ASAP7_75t_R cell. By comparing it with the transistor placement shown in Figure 4(b), it can be observed that the pseudo-pins of Pins a, b, and c are pruned to prevent potential design rule violations from occurring with transistors, and the pseudo-pins of Pin y exactly correspond to the diffusion locations.

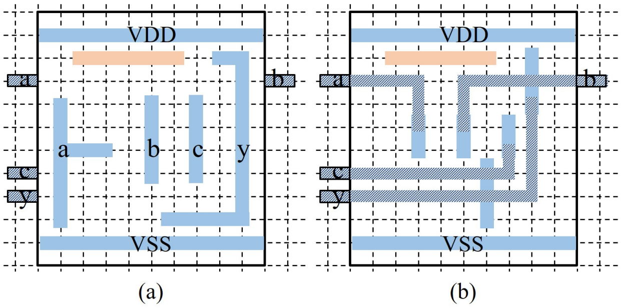

Figure 6: The concurrent detailed routing with original pin patterns and pseudo-pins. (a) An instance that cannot be successfully routed on Metal-1 with the original pin patterns. (b) The instance can be routed on Metal-1 by using the released routing resource from pseudo-pins.

4.2 Net Redirection

After completing the pseudo-pin extraction, multiple pseudo-pins are generated for each Type-1 connection. To ensure the connectivity among these pseudo-pins, we apply the minimum spanning tree algorithm to generate $k-1$ 2-pin nets fo k individual pseudo-pins, where the edges are weighted by their Manhattan distances. These additional 2-pin nets are taken into account during the concurrent detailed routing process.

4.3 Concurrent Detailed Routing

We develop our concurrent detailed router based on that in [5] while enabling routing resource maximization with pin pattern re-generation. The primary distinction in our router lies in the inclusion of two pin-related constraints in our formulation in addition to those in Section 2, which is detailed in the following.

4.3.1 Pseudo Pin Constraint. The pseudo-pins function similarly to the original pins, with the key distinction that unused access points are not wasted and can be utilized by other connections. Figure 5(a) presents two instances with four pins belonging to the two net labeled a and b. In Figure 5(c) shows the multi-commodity flow model derived from Figure 5(a), where the pins of Nets a and b are connected by super sources and super targets, and the vertices of the same net are indicated by the same color. If the original pin pattern is retained, even if the ILP formulation could theoretically yield an optimal solution, it is impossible to route the two nets by only using Metal-1 because the middle pins obstruct each other. However, by ensuring one of the access points on pseudo-pin patterns can be designated for pin access while allowing the remaining access points to be routed by another net, a flow solution can be found on the graph, as demonstrated in Figure 5(d). The final routing solution is then shown in Figure 5(b). Implementing this function is straightforward by adjusting the original obstacle constraint in Eq. (3). The set of obstacle vertices O^c initially encompasses different net original pin patterns and different net track assignments. In our implementation, we eliminate the different net original pin patterns from O^c to facilitate the routing of all connections on those vertices.

4.3.2 Characteristic Constraint. To minimize the impact on timing and power, it is desirable to achieve the closest electrical characteristics to the original cell library after pin pattern re-generation. Therefore, we restrict the Type-1 connections to be routed on Metal-1 exclusively. This restriction can be implemented straightforwardly by placing blockages on the unauthorized routing metal layers. We refine the constraint in Eq. (3) into that in Eq. (8). In this equation, PC includes all Type-1 pin connections, L^c represents the set of basic vertices within the layers that are forbidden for routing the connection c, and N_i^c denotes a set of neighboring vertices of v_i^c. For any $v_i^c \in L^c$, the total flow on the

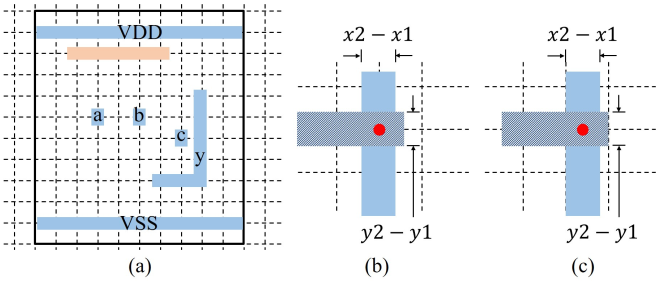

Figure 7: The pin re-generation process with different instance offsets. (a) The re-generated pin patterns from the result shown in Figure 6(b). (b) The center of the re-generated pin pattern on an on-track pin. (c) The center of the re-generated pin pattern on an off-track pin.

associate edges to v_i^c must be 0 to prevent the path of c from passing through v_i^c.

$$\sum_{v_a^c \in N_i^c} fe_{i,a}^c = 0, \forall v_i^c \in L^c, \forall c \in PC. \tag{8}$$

4.3.3 A Practical Example. To clarify the difference between our router and that in [5], we exhibit a practical example as shown in Figure 6, where Figure 6(a) gives a cell instance to be connected with four original pin patterns labeled a, b, c, and y. It can be observed that no feasible solution can be found by only using Metal-1 routing resources due to the occupation of fixed pin patterns. To implement the ideas proposed in this paper, the super sources and super targets in the flow model are connected to the pseudo-pin patterns shown in Figure 4(d). The pseudo-pin constraint removes the different net original pin patterns from the obstacle vertex set O^c to make sure the routing resource on Metal-1 can be fully used. Then, the characteristic constraint restricts the Type-1 connection between $y1$ and $y2$ to be routed on Metal-1. Finally, we apply the ILP solver to get an optimal solution, as demonstrated in Figure 6(b). It is noteworthy that the adoption of the pseudo-pins provides individual access points for pins $y1$ and $y2$, while the other access points facilitate the connections of nets c and b.

4.4 Pin Pattern Re-Generation

During the concurrent detailed routing stage, as the nets within a cluster obtain optimal solutions, these solutions can then be automatically transformed into re-generated physical pin patterns. In the case of a Type-3 connection, the access point can be directly connected to the corresponding gate or diffusion on the transistor placement through a single contact. This process involves locating the center of the access point and establishing a pin pattern with the minimum required area. To ensure alignment with the transistor placement, the center position is determined by considering both the pseudo pin pattern and the routed wire segment. Specifically, the x-coordinates of the pseudo pin boundaries, denoted as $pseudopin_{x1}$ and $pseudopin_{x2}$, along with the y-coordinates of the routed wire segment, labeled as $segment_{y1}$ and $segment_{y2}$, are utilized in the calculation of the center position, as shown in Eq. (9). It is important to note that the pseudo-pin patterns may not always align directly with the routing tracks due to standard cell offsets. However, regardless of whether the pin pattern aligns with the track or not, the re-generated pin pattern is positioned at the center coordinates that align with both the pseudo pin pattern and the routed segment, as demonstrated by the red points in Figure 7(b) and Figure 7(c).

$$\begin{aligned} x_{center} &= pseudopin_{x2} + pseudopin_{x1}/2 \\ y_{center} &= segment_{y2} + segment_{y1}/2 \end{aligned} \tag{9}$$

Table 2: The routing results of PACDR [5] and our work for the ISPD-18 contest benchmarks.

Case name	PACDR [5]				Ours			
	ClusN	SUCN	UnSN	CPU	SUCN	UnCN	SRate	CPU
ispd_test1	1076	908	168	11	159	9	0.946	18
ispd_test2	18642	15141	3501	165	3297	204	0.942	295
ispd_test3	18058	14607	3451	157	3249	202	0.941	283
ispd_test4	22522	20458	2064	392	2020	44	0.979	478
ispd_test5	21167	20685	482	374	440	42	0.913	487
ispd_test6	31438	30795	643	505	573	70	0.891	588
ispd_test7	52198	50651	1547	932	1291	256	0.835	983
ispd_test8	52000	50464	1536	931	1287	249	0.838	994
ispd_test9	50822	49348	1474	768	1213	261	0.823	836
ispd_test10	51166	49394	1772	829	1415	357	0.799	886
Comp	-	-	-	1	-	-	0.891	1.319

For a Type-1 connection, the process involves connecting multiple pseudo pins and creating a pin pattern. This pin pattern can be determined using a shortest path algorithm on the routing solution obtained from the optimal result. Figure 7(a) illustrates the re-generated pin patterns from the result obtained from Figure 6(b). Notably, for the Type-3 connections corresponding to pins a, b, and c, the re-generated pin patterns occupy only the minimum required area. On the other hand, for the Type-1 connection of pin y, the pin pattern also establishes a connection between pseudo-pins $y1$ and $y2$.

5 EXPERIMENTAL RESULTS

We implemented our algorithmic framework using the C++ programming language and conducted experiments on a Linux machine equipped with a 3.5 GHz CPU and 128 GB of memory. To solve the concurrent detailed routing and pin pattern re-generation process modeled as a multi-commodity flow problem, CPLEX 20.1.0.0 [26] is used as the ILP solver. We enhanced computational efficiency by employing multi-threading with OpenMP [27]. The details of the experimental flow have been introduced in Section 3. Two experiments were conducted to analyze routing quality and cell characteristics after pin pattern re-generation, whose results are respectively given in Section 5.1 and Section 5.2.

5.1 Routing Quality Analysis

The first experimental result highlights the effectiveness of the proposed concurrent detailed routing with pin pattern re-generation technique. As mentioned in Section 3, we implement the PACDR [5] as our initial detailed router to obtain the initial routing results. The PACDR finds an optimal solution if it exists for each local region (defined as a "cluster" in [5]). Therefore, adopting the PACDR can accurately calculate how many originally unroutable local regions become routable by re-generating pin patterns. Note that we follow [5] to apply the concurrent routing approaches to the clusters with multiple connections (defined as "multiple clusters"). Each cluster with only a single connection is solved with A*-search. Table 2 shows the experimental results. In this comparison, the key metrics include ClusN (the number of multiple clusters), SUCN (the number of solvable clusters), UnSN (the number of unsolvable clusters), SRate (success rate calculated as $SUCN/(SUCN + UnSN)$), and CPU (runtime in seconds). As depicted in Figure 3, our workflow begins by running the PACDR to route all clusters. After that, we identify the unsolvable clusters and proceed to perform our concurrent detailed routing with pin pattern re-generation for these clusters. Therefore, the total number of SUCN and UnCN in the column of "Ours" equals the number of UnSN in the column of "PACDR [5]". In addition, since the placement and the global routing of the adopted benchmarks are re-performed in our work, the numbers of multiple clusters are different from and more than those reported in [5]. The results in the table show that the proposed approach can resolve 89% of the unsolvable clusters left from the PACDR, which only costs an average of 31% more runtime after PACDR. The results

Table 3: The cell characteristic comparison of original cell pin patterns and re-generated cell pin patterns for the ASAP7 library.

Case name	Original Cell Pin Pattern								Re-generated Cell Pin Pattern							
	LeaakP	InterP	Trans	RNCap	RXCap	FNCap	FXCap	M1U	LeaakP	InterP	Trans	RNCap	RXCap	FNCap	FXCap	M1U
TIEHIx1	0.876	-	-	-	-	-	-	0.0143	0.8768	-	-	-	-	-	-	0.0141
INVx1	53.325	0.4604	441.3	0.4573	0.6437	0.4592	0.6411	0.0162	53.325	0.3965	440.4	0.4312	0.6161	0.4334	0.6135	0.0087
NAND2xp33	36.452	0.2273	627.2	0.2719	0.3672	0.2642	0.4062	0.0224	36.452	0.2216	626.5	0.2457	0.3419	0.2375	0.3425	0.0138
AOI21xp5	92.358	0.4879	428.5	0.4278	0.5838	0.4303	0.6058	0.0269	92.358	0.5050	428.5	0.4111	0.5684	0.4132	0.5898	0.0192
AOI211xp5	108.043	0.5903	614.7	0.3602	0.5299	0.3693	0.5267	0.0336	108.043	0.5681	615.0	0.3514	0.5179	0.3607	0.5150	0.0239
AOI221xp5	109.066	0.6448	609.6	0.3655	0.5312	0.3707	0.5308	0.0359	109.066	0.6472	609.6	0.3553	0.5186	0.3602	0.5202	0.0290
AOI33xp33	112.541	0.6597	618.8	0.3680	0.5175	0.3644	0.5203	0.0425	112.541	0.6538	618.8	0.3542	0.5069	0.3506	0.5080	0.0318
AOI322xp5	141.018	0.8915	617.2	0.3690	0.5785	0.3703	0.5989	0.0491	141.018	0.8911	617.2	0.3588	0.5684	0.3599	0.5886	0.0397
AOI332xp33	167.643	1.0380	619.6	0.4243	0.6106	0.4226	0.6108	0.0534	167.643	1.0310	619.6	0.4132	0.5996	0.4122	0.5977	0.0422
AOI333xp33	169.177	1.1650	625.5	0.4243	0.6102	0.4227	0.6094	0.0593	169.177	1.1450	625.5	0.4132	0.5979	0.4122	0.5973	0.0468
Comp	1	1	1	1	1	1	1	1	1	0.9782	0.9997	0.9597	0.9710	0.9595	0.9610	0.7516

show that applying the proposed pin re-generation technique is very effective in solving pin access-induced detailed routing hotspots.

5.2 Cell Characteristics Analysis

The second experiment focuses on the characterization of standard cells using both the original pin patterns and the re-generated pin patterns. Various metrics are considered, including LeakP (the maximum leakage power in pW), InterP (the maximum internal power in pW), and Trans (transition delay in pS). Additionally, the analysis also reports RNCap (the minimum rise capacity in fF), RXCap (the maximum rise capacity in fF), FNCap (the minimum fall capacity in fF), FXCap (the maximum fall capacity in fF), and M1U (Metal-1 usage in μm^2) for all pin patterns. Table 3 shows the results. One of the notable distinctions observed in the re-generated pin patterns is significant differences in pin capacitances. The re-generated pin patterns are considerably shorter than the original pin patterns, resulting in lower rise and fall capacitances. On average, the rise/ fall capacitance decreases by 3%. The transition delay that is also influenced by pin capacitance is thus slightly decreased. In addition, the total Metal-1 usage for pin patterns is reduced by an average of 24% compared to the original pin patterns, which directly contributes to much better routability as demonstrated in the first experiment. Regarding power characteristics, the leakage power remains consistent with those with original pin patterns, and the internal power averagely decreases by 2%.

6 CONCLUSIONS

This paper proposes the first work in the literature on concurrent detailed routing with pin pattern re-generation, which serves as a novel solution to the difficult pin access problem in the conventional cell-based design flow. Since the fixed pin patterns in standard cells may actually waste Metal-1 routing resources, we propose to re-route the unroutable local regions again by leveraging the power of pin pattern re-generation. The idea is realized by first applying pseudo-pin extraction from diffusions or gate areas and net redirection to collect required connections by pin patterns. After that, the concurrent detailed routing phase enhances an existing ILP formulation with the pseudo-pin constraint and the characteristic constraint to optimize resource utilization and attain superior routing solutions. The flow was implemented using the ASAP7_PDK cell library and the ispd_18 benchmark design, demonstrating a high success rate in re-routing local regions unroutable with the original pin patterns. In addition, we verified that pin pattern re-generation does not greatly impact the timing and power performance of cells and thus can be practically applied in real designs.

REFERENCES

[1] M. Ahrens et al., "Detailed Routing Algorithms for Advanced Technology Nodes, " in *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems (TCAD)*, vol. 34, no. 4, pp. 563-576, Apr. 2015.

[2] C.-K. Cheng, D. Lee and D. Park„ "Standard-Cell Scaling Framework with Guaranteed Pin-Accessibility," in *International Symposium on Circuits and Systems (ISCAS)*, pp. 1-5 2020.

[3] P. V. Cleeff, S. Hougardy, J. Silvanus and T. Werner, "BonnCell: Automatic Cell Layout in the 7-nm Era, " in *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems (TCAD)*, vol. 39, no. 10, pp. 2872-2885, Oct. 2020.

[4] C.-T. Ho, A. Ho, M. Fojtik, M. Kim, S. Wei, Y. Li, B. Khailany, and H. Ren, "NVCell 2: Routability-Driven Standard Cell Layout in Advanced Nodes with Lattice Graph Routability Model, " in *International Symposium on Physical Design (ISPD)*, pp. 44-52, 2023.

[5] Y.-J. Jiang and S.-Y. Fang, "Pin Access-Oriented Concurrent Detailed Routing, " in *International Symposium on Physical Design (ISPD)*, pp. 17-25, 2023.

[6] A. B. Kahng, L. Wang and B. Xu, "The Tao of PAO: Anatomy of a Pin Access Oracle for Detailed Routing," in *Design Automation Conference (DAC)*, pp. 1-6, 2020.

[7] D. Lee, D. Park, C.-T. Ho, I. Kang, H. Kim, S. Gao, B. Lin, C.-K. Cheng, "SP&R: SMT-Based Simultaneous Place-and-Route for Standard Cell Synthesis of Advanced Nodes," in *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems (TCAD)*, vol. 40, no. 10, pp. 2142-2155, Oct. 2021.

[8] Y.-L. Li, S.-T. Lin, S. Nishizawa, H.-Y. Su, M.-J. Fong, O. Chen and H. Onodera, "NCTUcell: A DDA- and Delay-Aware Cell Library Generator for FinFET Structure With Implicitly Adjustable Grid Map, " in *ACM/IEEE International Symposium on Low Power Electronics and Design (ISLPED)*, vol. 41, no. 12, pp. 5568-5581, Dec. 2022.

[9] M. M. Ozdal, "Detailed-Routing Algorithms for Dense Pin Clusters in Integrated Circuits, " in *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems (TCAD)*, vol. 28, no. 3, pp. 340-349, Mar. 2009.

[10] N. Ryzhenko, S. Burns, A. Sorokin, and M. Talalay, "Pin Access-Driven Design Rule Clean and DFM Optimized Routing of Standard Cells under Boolean Constraints, " in *International Symposium on Physical Design (ISPD)*, pp. 41–47, 2019.

[11] J. Seo, Jinwook Jung, Sangmin Kim and Youngsoo Shin, "Pin accessibility-driven cell layout redesign and placement optimization, " in *Design Automation Conference (DAC)*, pp. 1-6, 2017.

[12] H. -Y. Su, S. Nishizawa, Y. -S. Wu, J. Shiomi, Y. -L. Li and H. Onodera, "Pin accessibility evaluating model for improving routability of VLSI designs, " in *International System-on-Chip Conference (SOCC)*, pp. 56-61, 2017.

[13] F. Wang, J. Liu, and E. F.Y. Young, "FastPass: Fast Pin Access Analysis with Incremental SAT Solving," in *International Symposium on Physical Design (ISPD)*, pp. 9-16, 2023.

[14] X. Xu, B. Cline, G. Yeric, B. Yu and D. Z. Pan, "Self-Aligned Double Patterning Aware Pin Access and Standard Cell Layout Co-Optimization," in *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems (TCAD)*, vol. 34, no. 5, pp. 699-712, May 2015.

[15] X. Xu, B. Yu, J. -R. Gao, C.-L. Hsu and D. Z. Pan, "PARR: Pin-Access Planning and Regular Routing for Self-Aligned Double Patterning, " in *ACM Transactions on Design Automation of Electronic Systems (TODAES)*, vol. 21, no. 42, pp. 1-21, Jul. 2016.

[16] X. Xu, Y. Lin, V. Livramento and D. Z. Pan, "Concurrent pin access optimization for unidirectional routing, " in *Design Automation Conference (DAC)*, pp. 1-6, 2017.

[17] W. Ye, B. Yu, D. Z. O. Pan, Y.-C. Ban, and L. Liebmann, "Standard Cell Layout Regularity and Pin Access Optimization Considering Middle-of-Line," in *Great Lakes Symposium on VLSI (GVLSI)*, pp. 284-294 2015.

[18] J. P. Duarte, S. Khandelwal, A. Medury, C. Hu, P. Kushwaha, H. Agarwal, A. Dasgupta, and Y. S. Chauhan, "BSIM-CMG: Standard FinFET compact model for advanced circuit design," in *European Solid-State Circuits Conference (ESSCIRC)*, pp. 196-201, 2015.

[19] S. Mantik, G. Posser, W. Chow, Y. Ding, and W. Liu, "ISPD 2018 Initial Detailed Routing Contest and Benchmarks," in *International Symposium on Physical Design (ISPD)*, pp. 140–143, 2018

[20] L. T. Clark, V. Vashishtha, L. Shifren, A. Gujja, S. Sinha, B. Cline, C. Ramamurthy and G. Yeric, "ASAP7: A 7-nm finFET predictive process design kit," in *Microelectronics Journal (MICROJ)*, vol. 53, pp. 105-115, July 2016.

[21] The open source detailed router TritonRoute-WXL,
https://github.com/ABKGroup/TritonRoute-WXL

[22] Cadence Meet PPA and TAT targets at advanced nodes Innovus Implementation System,
https://www.cadence.com/en_US/home/tools/digital-design-and-signoff/soc-implementation-and-floorplanning/innovus-implementation-system.html

[23] Siemens IC sign-off verification and DFM optimization Calibre Design Solutions,
https://eda.sw.siemens.com/en-US/ic/calibre-design/

[24] Synopsys Accurate libraries for signoff & implementation SiliconSmart,
https://www.synopsys.com/implementation-and-signoff/signoff/siliconsmart.html/

[25] Synopsys PrimeSim HSPICE,
https://www.synopsys.com/implementation-and-signoff/ams-simulation/primesim-hspice.html/

[26] IBM High-performance optimization solver for linear, mixed-integer and quadratic programming ILOG CPLEX Optimizer,
https://www.ibm.com/analytics/cplex-optimizer/

[27] The OpenMP API specification for parallel programming,
https://www.openmp.org/