

CoRe-ECO: Concurrent Refinement of Detailed Place-and-Route for an Efficient ECO Automation

Chung-Kuan Cheng¹, Andrew B. Kahng^{1,2}, Ilgweon Kang³, Minsoo Kim²,
Daeyeal Lee², Bill Lin², Dongwon Park² and Mingyu Woo²

¹Department of Computer Science and Engineering, University of California San Diego, La Jolla, USA

²Department of Electrical and Computer Engineering, University of California San Diego, La Jolla, USA

³Digital & Signoff Group, Cadence Design Systems Inc., San Jose, USA

{ckcheng, abk, mik226, ldaeyeal, billlin, dwp003, mwoo}@ucsd.edu, ikang@cadence.com

Abstract—With the relentless scaling of technology nodes, physical design engineers encounter non-trivial challenges caused by rapidly increasing design complexity, particularly in the routing stage. Back-end designers must manually stitch/modify all of the design rule violations (DRVs) that remain after automatic place-and-route (P&R), during the implementation of engineering change orders (ECOs). In this paper, we propose *CoRe-ECO*, a concurrent refinement framework for efficient automation of the ECO process. Our framework efficiently resolves pin accessibility-induced DRVs by simultaneously performing detailed placement, detailed routing, and cell replacement. In addition to perturbation-minimized solutions, our proposed SMT-based optimization framework also suggests the adoption of alternative master cells to better achieve DRV-clean layouts. We demonstrate that our framework successfully resolves from 33.3% to 100.0% (58.6% on average) of remaining DRVs on M1-M3 layers, across a range of benchmark circuits with various cell architectures, while also providing average total wirelength reduction of 0.003%.

Index Terms—pin accessibility, refinement, ECO, SMT

I. INTRODUCTION

With the relentless scaling toward advanced technology nodes, increasingly sophisticated IC fabrication constraints (e.g., fewer routing tracks, higher pin density, and complicated conditional design rules) bring rapidly increasing design complexity [16]. This leads to non-trivial challenges for physical design, particularly in the routing stage. The number of remaining design rule violations (DRVs) after place-and-route (P&R) has become one of the most crucial metrics for an automatic IC layout solution, since back-end designers must manually stitch/modify all DRVs via implementation of engineering change orders (ECOs) at the post-layout stage. In particular, achieving the pin accessibility needed to resolve DRVs is a critical, time-consuming engineering task just before tapeout, and is therefore a critical bottleneck in the advanced-node IC design process [7], [18].

To mitigate pin accessibility-induced DRVs, several approaches are proposed to improve pin accessibility through detailed placement (DP) optimization [6], [8], [9], [12], [15], [20] and standard cell layout optimization [3], [19]. The

authors of [12] perform DP optimization using a global routing solution as guidance, with pin accessibility modeled only in the form of pin density. Dynamic programming and deep learning-based DP optimizations considering each pin’s access are developed in [6], [8], [9], [20]. In [15], the authors introduce a measurement of inaccessible pins in a cell to optimize DP. However, these models have limited capability of comprehending design rules in detailed routing (DR). The works of [3] and [19] have proposed pin accessibility-driven cell layout optimization frameworks for improving routability at block-level. Recently, the authors of [11] perform replacement of inaccessible cells with diverse cell layouts in terms of pin locations and number of access points, in the ECO stage. The applicability of these works is intrinsically limited due to the lack of holistic consideration for the block-level design steps (e.g., DP and DR).

In this work, we propose *CoRe-ECO*, a **Concurrent Refinement** framework which simultaneously performs incremental DP and DR, along with cell replacement, at the **ECO** stage. To our knowledge, this is the first work to present a concurrent co-optimization of DP, DR, and cell replacement within block-level P&R. Our main contributions are summarized as follows.

- We propose a concurrent refinement framework which simultaneously performs DP, DR, and cell replacement for each local window (i.e., switchbox) covering DRV locations, to determine ECOs at post-layout stage. We devise a novel dynamic cell allocation (DCA), merging the refinement steps into a single-step optimization.
- *CoRe-ECO* performs (i) placement adjustment such as horizontal/vertical shifting, horizontal flipping, and cell swapping; (ii) pin-length extension with a recommendation of adopting alternative master cells while maintaining the same functionality; and (iii) routing optimization to seek the best-quality DRV-clean solution.
- *CoRe-ECO* minimizes perturbation of the given layout by utilizing a satisfiability modulo theories (SMT) solver, enabling multi-objective optimization.
- We validate the proposed *CoRe-ECO* framework with various testcases, demonstrating successful fixing of pin accessibility-induced DRVs through the proposed ECO flow along with total wirelength optimization.

Research in Cheng and Lin groups is partially supported by NSF CCF-2110419 and NSF IIS-1956339. Research in ABKGroup is partially supported by DARPA HR0011-18-2-0032, NSF CCF-2112665, Samsung, Qualcomm, NXP Semiconductors, Mentor Graphics and the C-DEN Center.

TABLE I
NOTATIONS FOR THE PROPOSED *CoRe-ECO* FRAMEWORK.

Term	Description
T	Set of instances in a switchbox
t	t^{th} instance
l_t	0-1 indicator if instance t is flipped
x_t	x -axis coordinate of lower-left corner of t
y_t	y -axis coordinate of placement row of t
y_{org}^t	Initial y -axis coordinate of placement row of t
x_{org}^t	Initial x -axis coordinate of lower-left corner of t
w_t	Width of instance t
P^t	Set of internal pins of instance t
p_i^t	i^{th} pin of instance t
$a_{ext}(p_i^t)$	Set of extended vertices of pin p_i^t
$V(V_i)$	Set of vertices in (i^{th} metal layer of) the routing graph G
v	A vertex with the coordinate (x_v, y_v, z_v)
$a(v)$	Set of adjacent vertices of v
$e_{v,u}$	An edge between v and u , $u \in a(v)$
$w_{v,u}$	Weighted cost for metal segment on $e_{v,u}$
N	Set of multi-pin nets in the given routing box
n	n^{th} multi-pin net
f_m^n	A two-pin subnet connecting a source and sink, i.e., a commodity
$e_{v,u}^n$	0-1 indicator if $e_{v,u}$ is used for n
$f_m^n(v,u)$	0-1 indicator if $e_{v,u}$ is used for commodity f_m^n
$m_{v,u}$	0-1 indicator if there is a metal segment on $e_{v,u}$
$C_m^n(v,u)$	Capacity variable for $e_{v,u}$ of commodity f_m^n

The remaining sections are organized as follows. Section II describes the proposed *CoRe-ECO* framework. Section III discusses our experimental setup and results. Section IV concludes the paper.

II. CORE-ECO FRAMEWORK

This section introduces an overview of the proposed framework, grid-based P&R architecture, refinement operations, switchbox generation, perturbation-minimized optimization, and SMT formulation.

A. Framework Overview

We formulate a conventional (sequential) layout refinement process as a constraint satisfaction problem (CSP) with variables and constraints to integrate placement adjustment and routing steps into a single multi-objective optimization problem. We adopt the SMT solver Z3 [5] to solve the given optimization problem. Fig. 1 illustrates an overview of our framework. Given standard cell library, switchbox, and instance slack information,¹ our framework simultaneously obtains the optimal solution that strictly satisfies the constraints integrated into our novel DCA scheme. Our notations are described in Table I.

B. Grid-based Place-and-Route Architecture

We define the grid-based placement and 3-D routing graph composed of four metal layers (i.e., $M1$ - $M4$) as shown in Fig. 2. Cell instances and I/O pins are aligned with $M1$ vertical tracks and gate poly of the standard cell. Inspired by [10], we adopt supernodes to cover the multiple candidates for each pin, either the I/O pin of a standard cell (i.e., P_{IN}) or the outer pin of a switchbox (i.e., P_{EX}). The location of P_{IN} is dynamically determined by placement formulation and is associated with the flow formulation for routing through DCA. P_{EX} interconnects the internal pins inside the switchbox to the

¹We define *instance slack* as the worst slack among pins of a given instance.

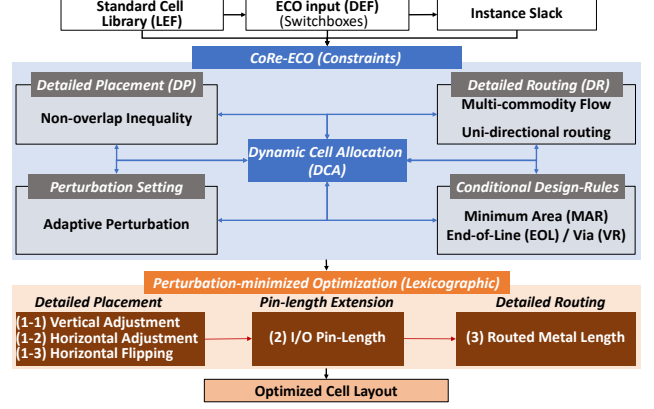


Fig. 1. Framework Overview.

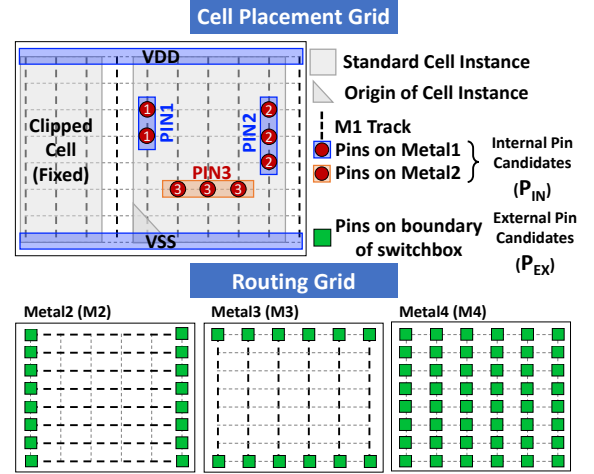


Fig. 2. Grid-based Place-and-Route Architecture.

outer pins and is located along the boundary of the switchbox which corresponds to the pre-routed result.

The horizontal routing grid (i.e., $M2$ and $M4$) consists of eight tracks per placement row and the vertical routing grid (i.e., $M3$) is aligned with the cell placement grid². Note that we focus on $M2$ - $M3$ layers assuming that $M2$ and $M3$ have the same metal pitches, because our proposed framework targets the correction of the pin accessibility-induced DRVs. Thus, $M4$ layer only contains VIA34 elements as the external pins for connections to the upper layer.

C. Refinement Operations

CoRe-ECO uses placement adjustment and pin-length extension as refinement operations within its adaptive perturbation method.

Placement Adjustment. Fig. 3 illustrates possible adjustments during DP to solve the pin accessibility-induced DRV. When the given placement layout in a switchbox (Fig. 3(a)) does not have feasible routing solutions due to an inaccessible $M1$ pin, *CoRe-ECO* adjusts the placement of instances in the switchbox by horizontal/vertical shifting, horizontal flipping, and cell

²We assume an on-grid routing scheme for each routing layer, consistent with sub-7nm multi-patterning technologies and IC practitioners' restriction of preferred routing direction per each layer [21].

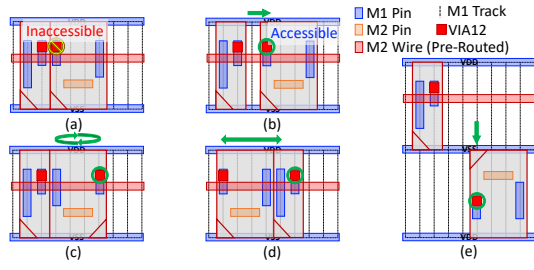


Fig. 3. Placement Adjustment. (a) Initial Placement. (b) Horizontal Shifting. (c) Horizontal Flipping. (d) Cell Swapping. (e) Vertical Shifting.

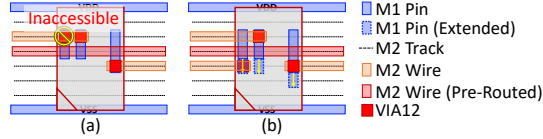


Fig. 4. Pin-length Extension. (a) Initial Placement. (b) Pin-length Extension.

swapping as shown in Figs. 3(b)-(e). Note that cell swapping is only performed between two adjacent instances.

Pin-length Extension. As a rule, master cells with minimum pin-length are preferred for use during initial P&R, e.g., to achieve better timing optimization. However, in the ECO stage, engineers should consider adopting alternative master cells for specific instances, so as to improve pin accessibility or to achieve the target design specification. While works of 20+ years ago [1], [2], [13] pursued *liquid library* approaches, today it is well-understood that a library cell must be qualified before it is used in a production chip. Thus, our framework suggests minimum-achievable pin-length extensions needed to fix DRVs: by only extending the metal segments of I/O pins, we enable engineers to adopt (i.e., swap in) alternative master cells that maintain the same functionality while minimizing the magnitude of undesired timing impact on each cell. Furthermore, our framework minimizes the effective number of alternative master cells by the perturbation-minimized optimization described in Section II-E. Note that this simple pin-extension satisfies all the conditional design rules for generating standard cell libraries that are described below in Section III-A. Fig. 4 illustrates pin-length extension that resolves a DRV caused by the inaccessible pin (Fig. 4(a)) by extending the instance’s I/O pins (Fig. 4(b)) without DP adjustments. We only apply the pin-extension for *M1* pins because (i) we do not allow routing on *M1* layer, and (ii) *M2* pins are directly accessible from BEOL layers.

Adaptive Perturbation. We utilize instance timing slacks to set the perturbation range (i.e., the range of the vertical and horizontal adjustments) by considering timing margins of each instance. For the instances with the worst slacks, we fix the placement and the routed wires to prevent the deterioration of timing characteristics. For the rest of the instances, the applicable range of perturbation is set by the input parameter settings. Flipping of instances and extension of pin-lengths are allowed for all instances except for the fixed instances.

D. Switchbox Generation

The proposed *CoRe-ECO* generates a switchbox by extracting the instance, pin, net, and obstacle information from

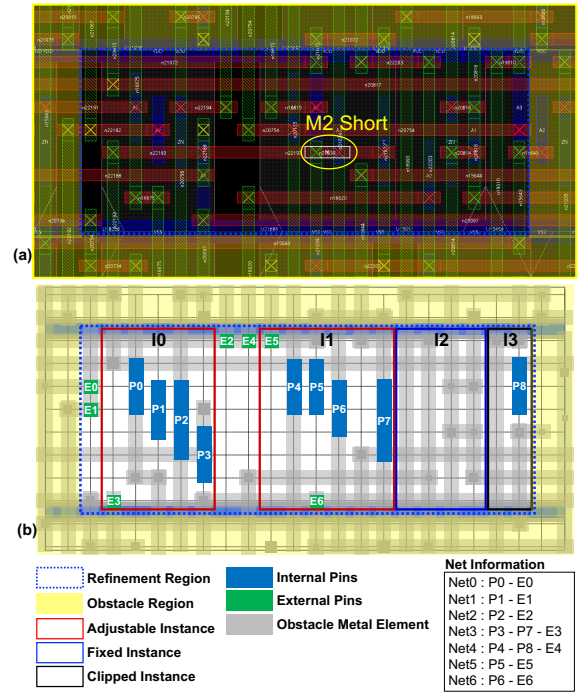


Fig. 5. Switchbox Generation. (a) Local window with 1 DRV, displayed by the commercial tool [22]. (b) Visualization of the generated switchbox.

each local window covering DRVs. Fig. 5(b) visualizes the generated switchbox representation from the local window depicted in Fig. 5(a).

Instances. We separate the instances in the refinement region (i.e., blue dotted box) according to the given P&R results. We first fix the placement and the corresponding pre-routed nets of the instances (i.e., I2) whose timing slacks are less than a predefined worst-slack upper bound. Then, the instances inside the region are extracted as adjustable instances (i.e., I0, I1) that are allowed for the refinement operations. The clipped instances (i.e., I3) are partially included in the refinement region. Therefore, those instances are not adjustable, but their partial pins/nets are extracted for the routing optimization.

Pins/Nets. The I/O pins in the adjustable and clipped instances are extracted as internal pin candidates (i.e., P0 - P8). The external pins (i.e., E0 - E6) are extracted along the boundary of the refinement region if there is a connection from the internal pins to the outside of the refinement region. The net information defines new interconnections between internal pins and external pins.

Obstacle Elements. The switchbox has two types of obstacle elements (i.e., gray rectangles). First, we extract the routed metal elements in the obstacle region (i.e., yellow solid box) and set those elements as obstacles to check for DRVs on the boundary of the refinement region. Second, we consider the routed elements inside the refinement region as obstacles if those elements do not have any connections to the internal/external pins or they are connected to the fixed instances.

E. Perturbation-minimized Optimization

The proposed *CoRe-ECO* has multiple objectives associated with the refinement operations and routing problems. To

honor the given (initial) DP and DR solution, we minimize the perturbations made by refinement operations as well as the total metal length. The vertical adjustment (ΔV) and horizontal adjustment (ΔH) are respectively defined as the total amount of vertical and horizontal shifts of the adjustable instances as shown in (1) and (2). The horizontal flipping (ΔF) is defined as the total number of flipped instances as shown in (3). The pin-length extension (ΔP) is defined as the sum of extended pin-lengths as shown in (4). The routing (ML) is the sum of routed VIA/Metal elements (i.e., $VIA12$, $M2$, $VIA23$, and $M3$). Each element has the same weight in the calculation of ML because we separate the objective functions for each type of element as shown in (5). *CoRe-ECO* simultaneously optimizes these multiple objectives in light of the “lexicographic” order described in (6). In other words, the objectives are optimized according to the priority order given by **LexMin**; for each given objective, this effectively induces a single-objective optimization problem under the constraining condition that optimizes the higher-priority objectives.

$$\text{Vertical Adjustment } (\Delta V) : \sum_{t \in T} (|y_t - y_{org}^t|) \quad (1)$$

$$\text{Horizontal Adjustment } (\Delta H) : \sum_{t \in T} (|x_t - x_{org}^t|) \quad (2)$$

$$\text{Horizontal Flipping } (\Delta F) : \sum_{t \in T} l_t \quad (3)$$

$$\text{Pin-length Extension } (\Delta P) : \sum_{e_{p,r} \in E} (w_{p,r} \times m_{p,r}), \quad (4)$$

$$\forall p \in P^t, \forall t \in T, \forall r \in a_{ext}(p)$$

$$\text{Routing (ML)} \{ \#VIA12, \#M2, \#VIA23, \#M3 \} : \sum_{e_{v,u} \in E} m_{v,u} \quad (5)$$

E =Sets of each $VIA12$, $M2$, $VIA23$, and $M3$ Element

LexMin: (a) ΔV , (b) ΔH , (c) ΔF , (d) ΔP ,

(e) ML $\{(1) \#VIA12, (2) \#M2, (3) \#VIA23, (4) \#M3\}$ (6)

F. SMT Formulation

1) *Placement Formulation*: We utilize the conventional floorplanning approach (i.e., *Relative Positioning Constraint (RPC)*) for the placement problem [17]. All instance positions in a specific placement row can be represented by two RPCs as shown in Fig. 6(a). At least one of the two inequalities holds for each pair $t \neq s$ through the SMT expression described in Algorithm 1. The maximum adjustable column boundary of each instance t is determined by the placement row y_t due to the different composition of clipped instances in each row as shown in Fig. 6(b). These geometric conditions determine the position and the flip status of the instance.

2) *Dynamic Cell Allocation (DCA)*: Every pin in each instance has its corresponding flow capacity variable $C_m^n(p, r)$ for certain net n and commodity m on the corresponding vertices of the placement grid, according to the shape and relative position of the pin in the instance as well as the possible adjustment range of each instance (see Fig. 7(a)). When locations of instances are determined by the placement

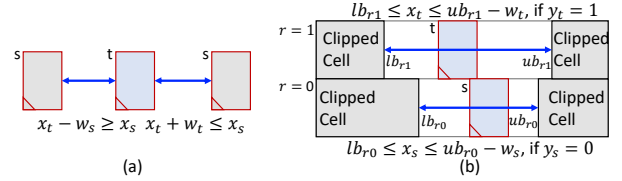


Fig. 6. Placement Constraints. (a) Relative positions between two instances in the same placement row. (b) Boundary condition of each placement row.

Algorithm 1 Set RPC Constraint (Instances t, s)

```

1: if  $y_t = y_s$  then ▷ t and s are on the same placement row
2:   if  $x_t >= x_s + w_s$  then
3:      $x_t >= x_s + w_s$ ; ▷ t is on the right side of s
4:   else if  $x_t + w_t < x_s$  then ▷ t is on the left side of s
5:      $x_t + w_t < x_s$ ;
6:   else
7:     Unsatisfiable condition;
8:   end if
9: end if

```

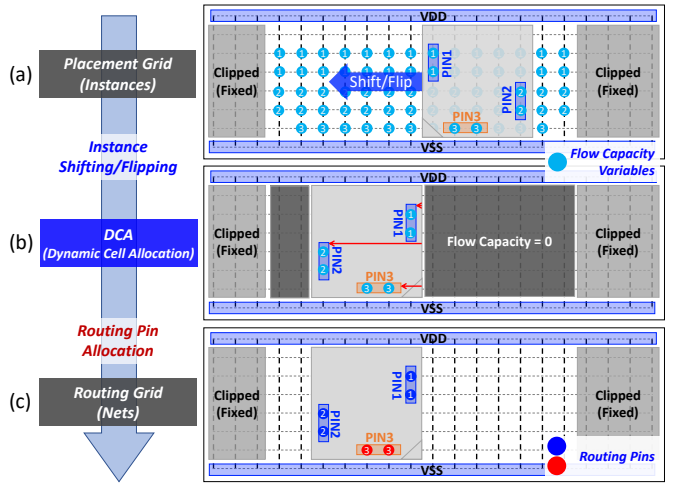


Fig. 7. Dynamic Cell Allocation (DCA).

formulation, the flow capacity variables of each instance’s pins are conditionally assigned to the corresponding locations according to the placement status of each instance (i.e., shifted, flipped) as described in Algorithm 2. First, the coordinates of each pin are determined by the location of each instance and flip status (Lines 1–6). Then, all capacities $C_m^n(p, r)$ outside the range of each pin are assigned to zero (Lines 7–11) as depicted in Fig. 7(b).

$$f_m^n(v = p, u = r) \leq C_m^n(p, r), \quad \forall r \in a(p), \forall r \in V_0 \quad (7)$$

Equation (7) associates the flow variable $f_m^n(v, u)$ with the flow capacity variable $C_m^n(p, r)$. Each $f_m^n(v, u)$ is determined by the routing formulation when vertex v is the internal pin p , and the adjacent vertex u is the adjacent vertex r of p in $M1, M2$ (i.e., V_1, V_2). This enables our routing formulation to recognize the feasible sets of r in V_1, V_2 layers as routing pins, as depicted in Fig. 7(c).

3) *Pin-length Extension*: We generate flow variables for extendable pin candidates to enable pin-length extension when finding a routable solution, as illustrated in Fig. 8. The extendable pins are generated in both up/down directions from the

Algorithm 2 Set Flow Capacity Control Constraint ($C_m^n(p, r)$)

```

/* x coordinate (resp. placement row) of a routing grid r:  $x_r$  (resp.  $y_r$ ) */
/* x coordinate (resp. placement row) of a pin p:  $x_p$  (resp.  $y_p$ ) */
/* p is either source or sink of a net n and commodity m */
/* origin's column, origin's row, flipping of a instance i:  $i_x, i_y, i_f$  */
/* x offset from the instance origin of a pin p:  $o_{x_p}$  (resp.  $o_{y_p}$ ) */
1: if  $i_f = \text{False}$  then
2:    $x_p = i_x + o_{x_p}$ ;
3: else
4:    $x_p = i_x - o_{x_p}$ ;
5: end if
6:  $y_p = i_y + o_{y_p}$ ;
7: if  $(x_r \neq x_p) \vee (y_r \neq y_p)$  then
8:    $C_m^n(p, r) = 0$ ;
9: else
10:   $C_m^n(p, r)$  is Determined by Routing Formulation;
11: end if

```

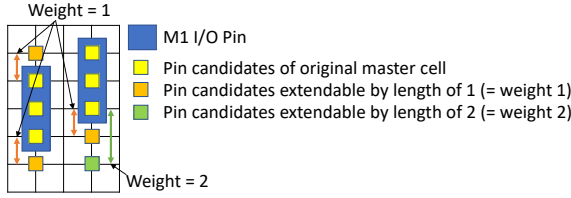


Fig. 8. Weighted extendable pin candidates.

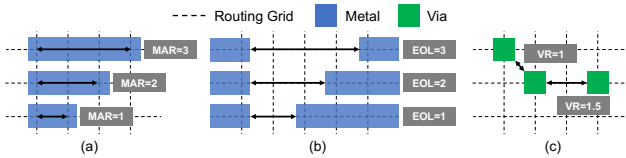


Fig. 9. Grid-based conditional design rules. (a) MAR, (b) EOL, (c) VR.

uppermost/lowermost vertex of each I/O pin. We set different weights for the extendable pin candidates, proportional to their distance from the nearest I/O pins. These weights are used as the priority in our objective function (Equation (4)) for minimizing the total length of the extended pins.

4) *Routing Formulation*: We use multi-commodity network flow and conditional design rules to formulate the DR problem, following the same principles as [14]. The flow formulation secures the routing path between the source and the sink for each commodity. Specifically, the refined constraints for *commodity flow conservation* and *vertex exclusiveness* in unidirectional edges are implemented in our framework to reduce the search space of the routing formulation. The conditional design rules work as constraints to route using design-rule violation-free paths. *CoRe-ECO* implements three fundamental grid-based design rules³, namely, *Minimum Area (MAR)*, *End-of-Line Spacing (EOL)* and *Via Rule (VR)*, as illustrated in Fig. 9. MAR (Fig. 9(a)) defines the minimum number of grids that should be covered by the metal segments. EOL (Fig. 9(b)) defines the minimum number of grids between two metal segments. VR (Fig. 9(c)) defines the minimum distance (in L_2 norm) between vias.

³In this work, we assume sub-7nm technologies that are based on Extreme Ultraviolet (EUV) lithography as in the previous work [4]. However, our framework is applicable to additional multi-pattern-aware design rules, such as Parallel Run Length (PRL) and Step Height Rule (SHR).

TABLE II
STANDARD CELL ARCHITECTURES.

Cell Architecture			Design Constraint	
#Fin	#Routing Tracks	Cell Height	Design Rule	Pin accessibility
3	6	8T	EUV-Loose (EL)	MPO3
3	6	8T	EUV-Tight (ET)	MPO3
2	4	6T	EUV-Loose (EL)	MPO2
			EUV-Tight (ET)	MPO3
			EUV-Tight (ET)	MPO2
			EUV-Tight (ET)	MPO3

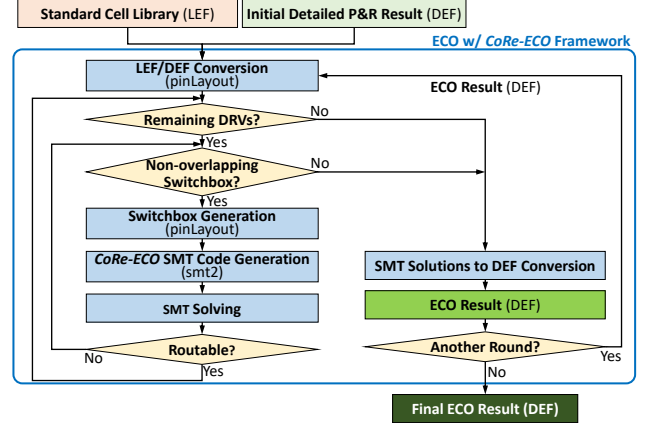


Fig. 10. Overall ECO flow using *CoRe-ECO* framework.

III. EXPERIMENTS

We have implemented the proposed *CoRe-ECO* framework in *Perl/SMT-LIB* 2.0 standard-based formula and validated on a Linux workstation with Intel (R) Xeon E5-2560L at 1.8GHz and 128GB memory. The SMT Solver Z3 (version 4.8.5) [5] is used to produce the optimized solution.

A. Experimental Environment

1) *Standard Cell Library Preparation*: Using an SMT-based cell layout automation [26], we prepare six types of standard cell libraries with various cell architecture and design constraints presented in Table II. We adopt two design rule sets that comprise combinations of specific design rule settings, as follows. EUV-Loose (EL) consists of MAR/EOL/VR = 1/1/1. EUV-Tight (ET) consists of MAR/EOL/VR = 1/2/1, inspired by [4]. We also generate two different types of cell libraries ensuring at least two and three I/O pin access points (i.e., MPO2 and MPO3). Then, for design enablement, we convert the primitive layout solutions of the SMT to LEF format. We assume the contacted poly pitch (CPP), metal pitch (MP), and cell height of 40, 40, and 280nm, respectively.⁴ We also generate three additional LEFs that have cells with pin-lengths extended by 1, 2 and 3 grids, respectively, consistent with the cell height of the corresponding standard cell library.

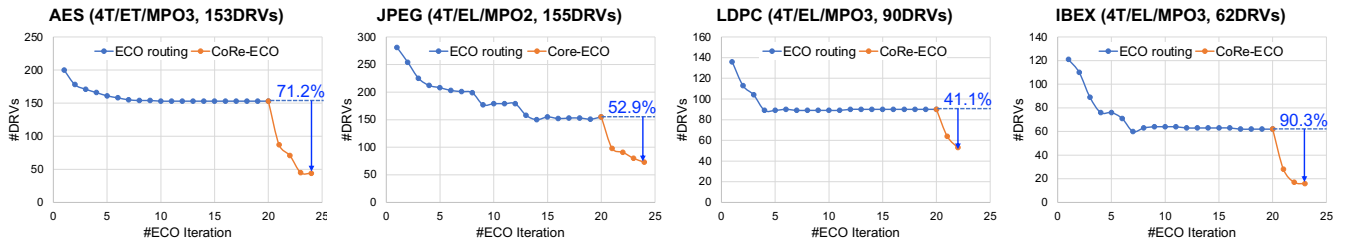
2) *Place-and-Route (P&R)*: We validate our framework by using four open-source RTL designs AES, JPEG, LDPC [24], and IBEX [23]. We utilize M2-M7 layers as BEOL (Back End of Line). We assume the power/ground pins on M1 layer for the initial detailed routing (DR). However, since our proposed framework targets grid-based architecture and correction of

⁴Since the layouts are fully grid-based, we consider the CPP and MP as pitches of grids.

TABLE III

EXPERIMENTAL STATISTICS. RT = THE NUMBER OF ROUTING TRACKS, DRSet = DESIGN RULE SET, MPO = MINIMUM PIN-OPENING PARAMETER, WL = TOTAL WIRELENGTH, Impr./Incr. = IMPROVEMENT/INCREMENT RATIO OVER ECO ROUTING, #V/#H/#F/#P = THE NUMBER OF REFINED INSTANCES BY VERTICAL SHIFTING/HORIZONTAL SHIFTING/HORIZONTAL FLIPPING/PIN-LENGTH EXTENSION, RUNTIME = OVERALL ECO FLOW RUNTIME.

Design		Cell Library			ECO routing		Proposed <i>CoRe-ECO</i> Refinement										
Name	#nets	#cells	RT	DRSet	MPO	#DRVs	WL(um)	#DRVs	Impr.	WL(um)	Incr.	#refinements				#ECO	Runtime
												#V	#H	#F	#P	Round	(h)
AES	13,958	13,694	6T	ET	3	40	46,272.16	2	95.0%	46,262.73	-0.020%	2	17	5	36	3	9.2
	13,912	13,648	4T	ET	2	74	45,968.76	13	82.4%	45,958.58	-0.022%	3	32	31	60	5	9.9
	13,938	13,674	4T	EL	3	41	45,797.96	0	100.0%	45,781.98	-0.035%	2	19	12	28	4	4.3
	13,802	13,538	4T	EL	2	146	45,035.54	56	61.6%	45,025.73	-0.022%	8	51	23	86	5	16
	13,867	13,603	4T	ET	3	153	44,548.12	44	71.2%	44,528.05	-0.045%	25	104	45	79	4	18.2
JPEG	70,543	70,518	6T	ET	3	79	144,905.98	30	62.0%	144,892.66	-0.009%	0	12	15	51	3	13.2
	71,491	71,466	4T	EL	2	155	134,901.91	73	52.9%	134,893.09	-0.007%	3	30	14	73	4	25.3
	71,177	71,152	4T	EL	3	37	133,558.22	18	51.4%	133,555.19	-0.002%	2	14	8	18	3	2.8
	70,932	70,907	4T	ET	2	198	132,739.50	132	33.3%	132,738.24	-0.001%	4	22	12	49	3	41.4
	70,008	69,983	4T	ET	3	68	136,852.58	42	38.2%	136,849.64	-0.002%	1	13	5	12	2	8.7
LDPC	57,133	55,081	6T	EL	3	21	732,917.12	8	61.9%	732,908.83	-0.001%	2	5	3	7	2	1.3
	57,138	55,086	4T	ET	2	12	737,957.04	7	41.7%	737,955.47	0.000%	0	3	1	1	1	0.24
	57,106	55,054	4T	EL	3	90	751,812.13	53	41.1%	751,797.69	-0.002%	7	19	7	9	2	5.3
IBEX	15,540	12,225	4T	EL	2	166	48,734.35	56	66.3%	48,730.00	-0.009%	12	40	16	80	5	15.8
	15,432	12,117	4T	EL	3	62	48,426.72	6	90.3%	48,425.94	-0.002%	3	11	9	8	3	4.8
	15,502	12,187	4T	ET	2	141	47,042.10	80	43.3%	47,045.31	0.007%	3	18	10	48	4	13.9
	15,179	11,864	4T	ET	3	93	49,346.74	32	65.6%	49,367.76	0.043%	0	9	14	13	3	5.1
Average						92.7	195695.11	38.4	58.6%	195689.23	-0.003%	4.5	24.6	13.5	38.7	3.3	11.5

Fig. 11. Example trends of the number of DRVs by the ECO routing and the proposed *CoRe-ECO* iterations.

the pin accessibility-induced DRVs, we focus on *M2-M3* layers assuming that *M2* and *M3* layers have the same metal pitches. Two commercial tools [22] [25] are used to generate the initial P&R layouts and to execute the following ECO routing. In the commercial tool, we perform 20 iterations of ECO routing until the commercial tool is unable to further reduce the number of DRVs (i.e., #DRVs) for most of the benchmark cases. The blue lines in Fig. 11 show the example trends of #DRVs through the ECO routing iterations for four representative cases from Table III (i.e., cases in bold). Note that we compare our work with the results of ECO routing because we are not able to fairly compare our work with the previous works, [6], [8], [9], [11], [12], [15], [20], due to (i) the different target design stage (i.e., DP optimization vs. ECO) and (ii) the lack of exact experimental settings.

3) *Setting up the Perturbation Range*: We set the 1% of worst slack cells and the routed nets connected to those cells as fixed instances and obstacle elements, respectively. For the rest of the instances, we set the perturbation range of the vertical and horizontal adjustments to two placement rows and eight poly pitches, respectively.

B. Design of Experiments

Fig. 10 illustrates an overview of the ECO flow utilizing our *CoRe-ECO* framework. Given a cell library and initial detailed P&R result, the new ECO round starts with converting these

layout information to a *pinLayout* format for the proposed framework in the **LEF/DEF Conversion** step. Then, if there exist any remaining DRVs in the non-overlapping regions, we rip up the region and generate a switchbox representation in the **Switchbox Generation** step. Note that the switchboxes in the same ECO round cannot overlap because our framework could change the P&R in each switchbox, and it also refers to the horizontal/vertical obstacle regions for checking the design rules on the boundary of each switchbox. Given the switchbox representation, we generate an SMT code and solve the problem through the **CoRe-ECO SMT Code Generation** and **SMT Solving** steps. We iterate these refinement steps until we find a routable solution or there are no remaining DRVs or feasible switchboxes in non-overlapping regions. After the iterations, in the **SMT Solutions to DEF Conversion** step, we apply the DRV-clean solutions to the original DEF and generate a revised DEF for the next ECO round or publish as the final ECO result.

The ECO flow described above is fully automated, and each sequence (i.e., switchbox generation to SMT solving) can be executed in parallel through the multi-threaded operation. In this work, up to 24 threads are used for all testcases. For each DRV, our framework examines multiple switchboxes of various sizes (i.e., 10 - 25 vertical tracks and 1 - 5 placement rows) for several relative locations to the target DRV. The size of a routable switchbox for each DRV varies according to the

TABLE IV

DETAILED EXPERIMENTAL RESULTS OF THE PROPOSED ECO FLOW (AES, 4T/ET/MPO2, 74 DRVs). #VTRACK/#ROW/#DRV/#NET/#PIN = THE NUMBER OF VERTICAL TRACKS/PLACEMENT ROWS/DRVs/NETS/PINS IN SWITCHBOX, /#TOTALINST./#ADJ.INST./#FIXEDINST. = THE NUMBER OF TOTAL/ADJUSTABLE/FIXED OR CLIPPED INSTANCES IN SWITCHBOX, #INST. = THE NUMBER OF PERTURBED INSTANCES, #V/#H/#F/#P = THE NUMBER OF REFINED INSTANCES BY VERTICAL SHIFTING/HORIZONTAL SHIFTING/HORIZONTAL FLIPPING/PIN-LENGTH EXTENSION.

Index	ECO Round	Switchbox							#Cell Refinement					SMT Runtime(s)
		#Vtrack	#Row	#DRV	#Total Inst.	#Adj. Inst.	#Fixed Inst.	#Net	#Pin	#V	#H	#F	#P	
0		18	2	1	5	5	2	15	33	0	2	0	1	11.0
1		19	1	2	4	3	1	9	22	0	0	0	2	1.3
2		18	2	1	4	3	1	13	29	0	1	1	1	8.5
3		14	1	1	3	2	1	6	17	0	0	0	1	0.6
4		14	1	2	3	2	1	6	15	0	0	0	0	0.6
5		21	2	1	7	3	4	15	43	0	0	0	3	25.4
6		17	1	1	3	3	0	7	17	0	2	0	1	3.4
7		15	2	1	4	2	2	9	21	0	1	0	0	28.8
8		14	1	1	3	2	1	8	17	0	0	0	1	0.7
9		14	1	1	2	0	2	6	14	0	0	0	0	0.4
10		18	2	1	8	5	3	16	41	0	0	3	3	11.8
11	1st	16	1	1	2	1	1	6	13	0	0	0	1	0.6
12		15	1	2	4	2	2	6	16	0	0	1	2	0.7
13		22	2	3	9	6	3	19	42	0	3	2	5	20.2
14		16	1	1	3	3	0	8	18	0	2	1	2	1.4
15		14	1	1	2	2	0	5	11	0	1	0	0	0.4
16		14	1	1	2	1	1	4	12	0	0	0	0	0.5
17		14	1	1	2	2	0	4	10	0	0	1	0	0.4
18		14	1	1	2	1	1	4	9	0	1	0	0	0.4
19		14	1	1	2	2	0	6	14	0	0	0	0	1.0
20		18	2	1	6	3	3	16	38	0	0	1	2	5.7
21		14	1	1	3	2	1	6	14	0	2	1	0	0.5
22		16	1	1	3	2	1	7	16	0	1	2	2	0.8
23		18	1	1	4	3	1	10	25	0	0	1	1	2.2
24		14	1	1	3	2	1	7	15	0	0	0	1	0.7
25		14	1	1	2	1	1	3	7	0	0	0	0	0.3
26	2nd	15	2	1	5	2	3	12	29	0	0	0	0	13.7
27		14	1	1	2	1	1	7	18	0	0	1	1	0.8
28		14	1	1	2	1	1	5	11	0	1	0	0	0.4
29		19	2	1	7	6	1	15	42	0	1	1	3	54.6
30		16	2	1	5	4	1	16	34	0	1	0	0	5.6
31		29	2	1	10	8	2	27	63	0	1	1	5	588.9
32		18	5	2	13	11	2	34	77	0	0	3	2	859.9
33		17	2	7	7	5	2	14	38	2	3	2	2	6.3
34	3rd	19	3	3	10	7	3	24	54	0	1	1	4	109.1
35		20	3	1	9	8	1	25	67	0	1	4	4	1465.8
36		14	3	3	7	5	2	16	41	0	0	0	1	10.8
37		17	2	1	5	2	3	18	39	0	1	0	2	8.3
38		18	5	3	13	7	6	33	71	1	2	2	3	1841.8
39	4th	14	1	2	2	2	0	5	12	0	2	1	1	0.6
40		21	2	1	9	6	3	16	43	0	0	1	2	30.7
41	5th	16	2	1	7	3	4	12	28	0	2	0	1	6.5
Average		16.6	1.7	1.5	5.0	3.3	1.6	11.9	28.5	0.1	0.8	0.7	1.4	122.2
Total				61	208	139	69	500	1196	3	32	31	60	5131.84

existing P&R results, routing congestion, and locations of the adjacent DRVs. To minimize the perturbation of the placement as well as the runtime of **SMT Solving** step, our framework increases the size of the switchbox from the minimum (i.e., 10 vertical tracks \times 1 row) to the maximum (i.e., 25 vertical tracks \times 5 rows) in **Switchbox Generation** step until it finds a routable solution or fails.

C. Experimental Results

1) *Statistics on the Proposed ECO Flow*: Table III summarizes the experimental statistics of the proposed ECO flow for benchmark cases which consist of four base design circuits synthesized with various cell libraries described in Table II. Column “ECO routing” represents the total number of DRVs (i.e., #DRVs) in target layers (i.e., $M2-M3$) and the total wirelength (i.e., WL) after the 20 iterations of ECO routing. The target DRVs mainly include “Cut Spacing” on $M1-M2$ layers and “Metal End-of-Line Spacing”, and “Metal Short” on $M2-M3$ layers. Our *CoRe-ECO* framework reduces the remaining DRVs after ECO routing by 58.6% on average, with reductions ranging from 33.3% to 100.0%. Fig. 11 shows the trend of #DRVs versus iterations of ECO routing

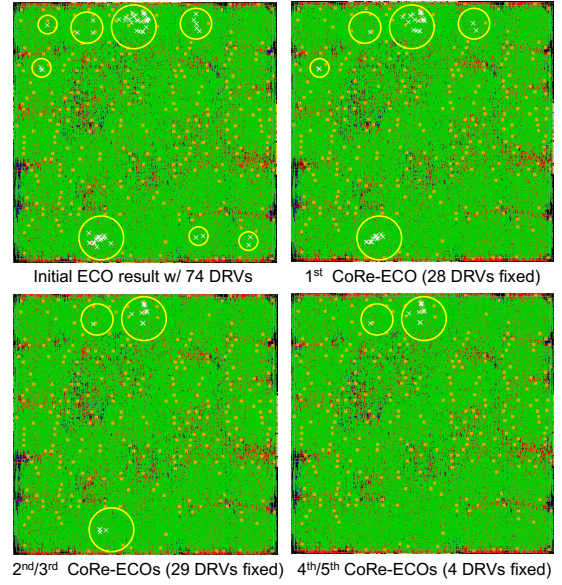


Fig. 12. DRV reductions by *CoRe-ECO* rounds for AES (4T/ET/MPO2).

(i.e., the blue line), along with the following *CoRe-ECO* flow (i.e., the orange line) for four representative cases of each base benchmark circuit from Table III (i.e., cases in bold). The figure demonstrates that our framework can further improve the routability with the concurrent cell refinements and the routing optimization. The reduction of the average total wirelength by 0.003% shows that our framework has successfully minimized the wirelength despite the refinement of cell placement and pin-length extension. We observe that the number of ECO rounds and the total runtime depend on #DRVs and the benchmark circuit configurations. For all benchmark cases, *CoRe-ECO* performed 3.3 ECO rounds on average with an average runtime of 11.5 hours.

Table IV presents the detailed refinement results of the AES benchmark circuit with 4T/ET/MPO2 cell library and 74 DRVs. The total of 5 ECO rounds with 42 switchboxes have been performed to resolve 61 out of all 74 DRVs. The average number of vertical tracks, placement rows, and DRVs in the switchboxes are 16.6, 1.1, and 1.5, respectively. Each switchbox includes 5.0 total/3.3 adjustable/1.6 fixed or clipped instances and 11.9 nets/28.5 pins on average. Through the 5 rounds of ECO flow, 126 out of 139 adjustable instances have been perturbed in the placement or the length of pins. 6 out of 42 ECO cases have been fixed without any perturbation. And 7 and 7 cases require the extension of pin-lengths or the replacement of instances, respectively. The remaining 22 cases are routable by only changing both instance placement and pin-lengths. The average runtime per switchbox is less than 3 minutes and the switchboxes up to 18×34 vertical/horizontal tracks, 7 adjustable instances, 33 nets, and 71 pins (i.e., Index 38) have been solved within 31 minutes. Fig. 12 shows the reduction of DRVs in full-chip layouts by multiple ECO rounds utilizing *CoRe-ECO* framework, displayed by a commercial tool [22]. The yellow circles indicate regions with DRVs.

2) *Example Refinement Operations*: Fig. 13 shows an example of refinement operations in our proposed ECO frame-

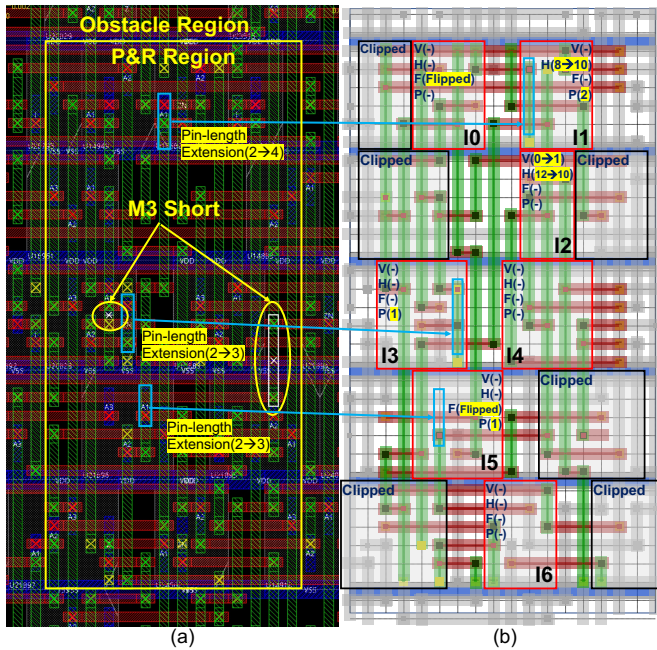


Fig. 13. Example of refinement operations in the proposed ECO flow (Index 38 case in Table IV). (a) Switchbox with 3 DRV's. (b) Routable solution with placement adjustments and pin-length extensions.

work. Fig. 13(a) depicts a switchbox of index 38 case in Table IV. The switchbox consists of 7 adjustable / 6 clipped (i.e. fixed) cell instances with 3 'M3 Short' DRV's in 18×34 vertical/horizontal tracks. Fig. 13(b) illustrates the DRV-clean solution with the refinement operations (i.e., placement adjustment and pin-length extension) and the optimized routing in terms of the metal length. Note that the elements in gray color represent the obstacles inside the switchbox and M1 I/O pins are not displayed in Fig. 13(b). The pre-routed wires, that (i) are connected to the fixed instances or (ii) have no internal connection inside the switchbox or (iii) exist outside the switchbox, are regarded as obstacles.

Placement Adjustment. The placement of the instance I2 in Fig. 13(b) has been adjusted from the placement row 0 to 1 and horizontally shifted from the vertical track 12 to 10. The instance I0 has shifted in the same placement row from the vertical track 8 to 10. Instances I0 and I5 have been flipped on the same placement locations.

Pin-length Extension. The I/O pins of the instance I1, I3, and I5 in Fig. 13(b) have been extended by 1–2 to maximize the pin accessibility. In the proposed ECO flow, the respective master cell of each of these instances is replaced with the additional master cell with extended pin-lengths, in the **SMT Solutions to DEF Conversion** stage.

IV. CONCLUSION

We have described a novel concurrent refinement framework for the automated ECO flow. Our framework provides simultaneous and perturbation-minimized refinements of DP-, DR-, and cell-optimized layout solutions to address the DRV's during the ECO stage. By ripping up and refining a local window of the whole layout design, *CoRe-ECO* is capable of

achieving a DRV-clean layout solution. We have demonstrated that our framework successfully resolves an average of 58.6% (range: 33.3% to 100.0%) of remaining post-ECO route DRV's on M1-M3 layers, across a range of benchmark circuits with various cell architectures, with no adverse effect on total routed wirelength (average of 0.003% reduction).

REFERENCES

- [1] D. Bhattacharya, V. Boppana, R. Roy and J. Roy, "Method for Automated Design of Integrated Circuits with Targeted Quality Objectives Using Dynamically Generated Building Blocks", *US Patent*, US7225423B2, 2007.
- [2] J. L. Burns and J. A. Feldman, "C5M-A Control-Logic Layout Synthesis System for High-Performance Microprocessors", *IEEE Trans. on CAD* 17(1) (1998), pp. 14–23.
- [3] C.-K. Cheng, C.-T. Ho, D. Lee and D. Park, "A Routability-Driven Complimentary-FET (CFET) Standard Cell Synthesis Framework Using SMT", *Proc. ICCAD*, 2020, pp. 1–8.
- [4] C.-K. Cheng, A. B. Kahng, H. Kim, M. Kim, D. Lee, D. Park and M. Woo, "PROBE2.0: A Systematic Framework for Routability Assessment from Technology to Design in Advanced Nodes", *IEEE Trans. on CAD* (2021), doi:10.1109/TCAD.2021.3093015.
- [5] L. De Moura and N. Bjørner, "Z3: An Efficient SMT Solver", *Proc. TACAS*, 2008, pp. 337–340.
- [6] Y. Ding, C. Chu and W.-K. Mak, "Pin Accessibility-Driven Detailed Placement Refinement", *Proc. ISPD*, 2017, pp. 133–140.
- [7] M.-K. Hsu, N. Katta, H. Y.-H. Lin, K. T.-H. Lin, K. H. Tam and K. C.-H. Wang, "Design and Manufacturing Process Co-Optimization in Nano-Technology", *Proc. ICCAD*, 2014, pp. 574–581.
- [8] A. B. Kahng, J. Kuang, W.-H. Liu and B. Xu, "In-Route Pin Access-Driven Placement Refinement for Improved Detailed Routing Convergence", *IEEE Trans. on CAD* (2021), doi:10.1109/TCAD.2021.3066528.
- [9] A. B. Kahng, L. Wang and B. Xu, "The Tao of PAO: Anatomy of a Pin Access Oracle for Detailed Routing", *Proc. DAC*, 2020, pp. 1–6.
- [10] I. Kang, D. Park, C. Han and C.-K. Cheng, "Fast and Precise Routability Analysis with Conditional Design Rules", *Proc. SLIP*, 2018, pp. 1–4.
- [11] S. Kim, K. Jo and T. Kim, "Boosting Pin Accessibility through Cell Layout Topology Diversification", *Proc. ASP-DAC*, 2021, pp. 1–6.
- [12] W.-H. Liu, C.-K. Koh and Y.-L. Li, "Optimization of Placement Solutions for Routability", *Proc. DAC*, 2013, pp. 1–9.
- [13] P. Majumder, B. Kumthekar, N. R. Shah, J. Mowchenko, P. A. Chavda, Y. Kojima, H. Yoshida and V. Boppana, "Method of IC Design Optimization via Creation of Design-Specific Cells from Post-Layout Patterns", *US Patent*, US7941776B2, 2011.
- [14] D. Park, D. Lee, I. Kang, C. Holtz, S. Gao, B. Lin and C.-K. Cheng, "Grid-Based Framework for Routability Analysis and Diagnosis with Conditional Design Rules", *IEEE Trans. on CAD* 39(12) (2020), pp. 5097–5110.
- [15] J. Seo, J. Jung, S. Kim and Y. Shin, "Pin Accessibility-Driven Cell Layout Redesign and Placement Optimization", *Proc. DAC*, 2017, pp. 54:1–54:6.
- [16] S. M. Y. Sherazi, M. Cupak, P. Weckx, O. Zografos, D. Jang, P. Debacker, D. Verkest, A. Mocuta, R. H. Kim, A. Spessot and J. Ryckaert, "Standard-Cell Design Architecture Options Below 5nm Node: The Ultimate Scaling of FinFET and Nanosheet", *Proc. SPIE 10962, Design-Process-Technology Co-optimization for Manufacturability XIII*, 1096202, 2019, pp. 1–16.
- [17] S. Sutanthavibul, E. Shragowitz and J. B. Rosen, "An Analytical Approach to Floorplan Design and Optimization", *IEEE Trans. on CAD* 10(6) (1991), pp. 761–769.
- [18] X. Qiu and M. Marek-Sadowska, "Can Pin Access Limit the Footprint Scaling?", *Proc. DAC*, 2012, pp. 1100–1106.
- [19] 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", *IEEE Trans. on CAD* 34(5) (2015), pp. 699–712.
- [20] T.-C. Yu, S.-Y. Fang, H.-S. Chiu, K.-S. Hu, P. H.-Y. Tai, C. C.-F. Shen and H. Sheng, "Pin Accessibility Prediction and Optimization with Deep Learning-Based Pin Pattern Recognition", *Proc. DAC*, 2019, pp. 1–6.
- [21] Cadence Design Systems, Inc., Sr Software Engineering Group Director, *personal communication*, 2019.
- [22] Cadence Innovus User Guide. <http://www.cadence.com>
- [23] lowRISC. <https://www.lowrisc.org>
- [24] OpenCores: Open-Source IP Cores. <https://opencores.org>
- [25] Synopsys IC Compiler II User Guide. <http://www.synopsys.com>
- [26] C.-K. Cheng, D. Lee and D. Park, "SMT-Based Standard-Cell Layout Generator". <https://github.com/ckchengucs/SMT-based-STDCCELL-Layout-Generator-for-PROBE2.0>