

Constraint-Driven Routing Closure for High-Frequency Mixed-Signal Designs

Rajashri Lakhota
Lattice Semiconductor
Pune, Maharashtra, India
rajashri.lakhota@latticesemi.com

Rohit Patel
Lattice Semiconductor
Pune, Maharashtra, India
rohit.patel@latticesemi.com

Krushna Kalamkar
Lattice Semiconductor
Pune, Maharashtra, India
krushna.kalamkar@latticesemi.com

Abstract - High-frequency mixed-signal designs place demands on routing that a standard congestion-driven digital router is not built to handle it produces jogs and sharp corners instead of straight paths, offers no built-in length matching across timing critical or differential groups, gives no direct control over shield wire spacing, and routes bus nets one at a time so that net order and pitch are lost. As a result, DRC-clean and signal-integrity-compliant routing is reached only through slow, repetitive manual work. A scripted, constraint-driven flow built on the top of Cadence NanoRoute High-Frequency (NRHF) router addresses these gaps by enforcing straight-line geometry across the declared routing layer range while meeting spacing, width, and enclosure rules, matching each net in a group to a target length using accordion, trombone, and coil styles, routing parallel and bus nets as a single group to preserve ordering and pitch, and placing shield wires under the same scripted control for predictable coupling and clean isolation between analog and digital domains. Measured at the first router pass on production designs, the flow reduced skew within matched net groups by 98%, routing-related DRC errors by 76%, and maximum coupled noise by 99%, with no cross-coupling violations in bus routing, while cutting manual routing effort by 95% with improved route quality.

Keywords - High-frequency routing, mixed-signal design, constraint-driven routing, length matching, signal integrity.

I. INTRODUCTION

Many SoC IPs impose explicit skew budgets on their interface signals. A DDR memory PHY must hold the data lines of each byte lane aligned to their strobe; a SerDes or PCIe lane must hold intra-pair skew small against the unit interval; MIPI D-PHY and JESD204B links specify lane-to-lane skew directly; and a high-speed converter's sampling clock must match its signal path. These budgets are stated in picoseconds, and routing is where they are met or lost.

Such designs demand more of routing than a congestion-driven digital router was built to give. That layout must follow circuit-level electrical constraints rather than wirelength alone was established early in the analog CAD literature [1], yet the default router still optimizes for

congestion, producing jogs and acute angles where straight geometry is needed for stable impedance.

Three gaps compound the problem. First, there is no native mechanism for equalizing delay across a matched group; length matching must satisfy minimum and maximum bounds inside the routing formulation [2], and applied late, dense meander segments introduce self-coupling that speeds propagation, so equal length no longer yields equal arrival [3]. Second, shielding and net ordering cannot be controlled explicitly; conventional global routing leaves up to 24% of nets violating RLC crosstalk limits at a 3 GHz clock [4]. Third, bus nets are routed sequentially, so ordering and pitch drift apart, although topology-matching bus routing is well defined [5].

DRC-clean, signal-integrity-compliant routing is therefore reached only through extensive manual intervention, which is slow, repetitive, and error-prone

II. LITERATURE REVIEW

Matching the electrical length of grouped nets has been studied as a routing-time problem rather than a post-processing fix. Ozdal and Hentschke formulated exact route matching for analog and mixed-signal circuits, proving structural properties of the problem and solving it with dynamic programming supported by heuristic search for faster runtime [6]. Yao, Cai, and Gao approached the same requirement from the detailed-routing side with LEMAR, a gridless single-layer router that partitions the layout into tiles, selects detour patterns according to tile shape, and applies an enhanced a search with backtracking and rip-up and reroute until the matching constraint is met [7]. More recent work extends matching to traces that are not confined to preferred directions: Fang et al. split the task into region assignment and meandering within the assigned region, and add multi-scale dynamic time warping so that differential pairs do not decouple while being meandered [8]. Together these works establish that matching quality depends on where the detour is placed, not only on how much length is added.

Shielding has been treated as a coupled decision problem rather than a fixed rule. Lepak, Luwandi, and He defined the simultaneous shield insertion and net ordering problem under an explicit RLC noise bound, using a conservative peaknoise model inside a simulated-annealing search, and

reported up to 71% fewer shields than greedy insertion for the same noise target [9]. Gao et al. applied the idea specifically to analog layout by classifying nets according to their performance requirement, translating that requirement into geometric properties, and inserting shield wires during a routing of critical nets while general and power nets are routed by a maze router [10]. Both results show that shield count, net order, and route order interact, and that treating shielding as a separate clean-up pass is wasteful.

A parallel line of work addresses how constraints are expressed and enforced. Pan et al. unified process rules and designer intent into a single configurable representation carried from schematic to layout through the OpenAccess constraint group format [11]. Chen et al. moved the problem toward production relevance with a detailed-routing framework that honors industrial design rules alongside analog geometric and electrical constraints [12]. SAGERoute handles geometric constraints such as design rules and low bending together with electrical constraints including electromigration, IR drop, and symmetry, using Steiner-tree-based wire sizing followed by guided detailed routing [13]. At the system level, MAGICAL builds a netlist-to-GDSII flow in which constraint extraction, device generation, placement, and routing are automated end to end [14]. A recent survey of VLSI routing confirms that analog, bus, and constraint-aware routing remain open problems, and that most published evaluation is carried out on academic benchmarks [15].

Each capability required by high-frequency mixed-signal closure therefore has mature prior art, but the literature partitions them. Matching, shielding, group routing, and constraint representation are optimized in separate stages and separate papers, mostly on research prototypes measured against contest benchmarks. What is not reported is how these requirements are held together simultaneously on a commercial router inside a production design, where the same net must satisfy straight-line geometry, a length target, a shielding rule, and a bus ordering rule at once, and where the practical baseline is manual engineering effort rather than an alternative algorithm. The present work addresses that gap.

III. METHODOLOGY

A. System level Introduction of Method

The methodology adds a scripted constraint layer above the Cadence NanoRoute High-Frequency router. Net groups are read from a specification file, compiled into routing constraints through `set_integration_route_constraint`, and enforced by the router itself. Pin shapes are extended to legal tracks before routing, and closure is driven by an automated check and refine loop.

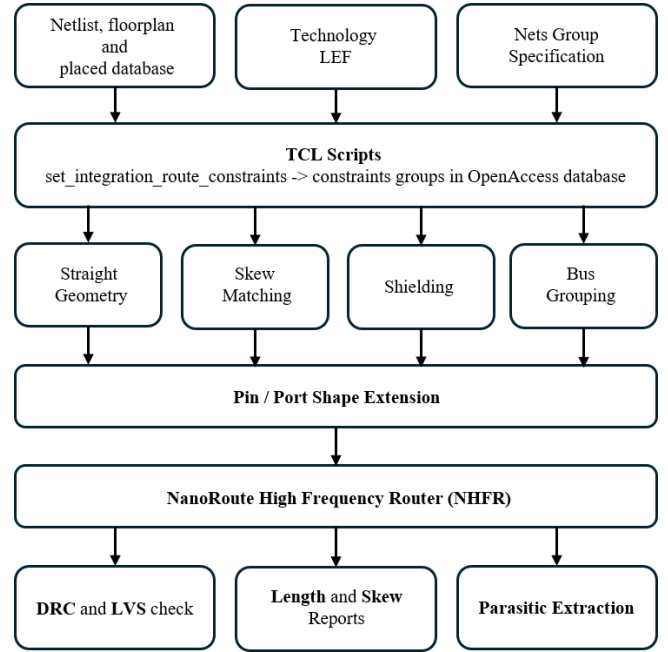


Figure 1. Constraint driven routing closure flow

Fig. 1 describes the structure of the flow. Three input streams, namely the placed database, the technology data with its track grid and non-default rules, and the net group specification, converge on a single scripted compilation layer, which is the only place where design intent is expressed. That layer emits four classes of constraint, drawn side by side because they are applied together rather than in sequence: straight geometry, length matching, shielding, and bus grouping. All four pass through pin and port shape extension before the router is invoked, so the router sees legal access shapes and a complete constraint set at the same time. The three verification paths at the bottom return to the constraint layer along the dashed arrows, which is what makes the flow a closure loop rather than a one-pass script.

B. Dataset Details

The dataset consists of production mixed-signal blocks taken at the placed, unrouted stage. Preprocessing extracts the net list and the constraints to be applied to it, assigns a group type, resolves the target length window from the longest member, aligns pin shapes to the track grid, and validates that every named supply exists before constraints are emitted.

TABLE I. TEST DESIGNS

Design	Node	Nets Under Constraints	Constraints applied
D1	28nm	2k	Match, Shielding (side), Bus
D2	28nm	5k	match, shield (coaxial)
D3	28nm	7k	bus, shield (dual-net side)

C. System Design and Implementation

The flow is a Tcl layer above the NanoRoute High-Frequency (NRHF) router in the Innovus Stylus Common UI. Intent is declared once per net group through the integration route constraint command, `set_integration_route_constraint`, which writes specialty routing constraints into the OpenAccess database, where the digital and custom environments both see them.

Straight geometry - Each group is given a straight access style, a controlled corner behavior, and a bounded, hard layer range. These remove the staircase segments and acute angles a congestion-driven search produces, keeping a uniform cross section and predictable impedance.

Length matching - The groups that carry a skew budget come from the IP interface definition, not from layout judgment: the byte lanes and strobe of a memory PHY, the lanes of a serial link, the clock and signal paths of a converter. Each is declared with a target length window and a tolerance derived from that budget. The router absorbs the difference between shortest and longest member using accordion, trombone, or coil compensation, and the match can be forced to hold layer by layer rather than on total length, which matters when members use different stacks. For a group G with target L_t and tolerance τ ,

$$Li_{eff} = Li_{route} + \Delta Li_{comp} + \delta i_{ext} \quad (1)$$

$$S(G) = \max_{G \in Li_{eff}} - \min_{G \in Li_{eff}} \leq (\tau/100) \cdot L_t \quad (2)$$

where ΔLi_{comp} is the length absorbed by the compensation structure and δi_{ext} is the contribution of the extended pin shape.

Shielding - Shielding is a declared property of a net group, not a clean-up pass, and three configurations are used. Sensitive nets and analog groups take side shields on both sides, tied to the reference supply, with gap and width set per layer. Where a net crosses noisy digital territory, a coaxial configuration also encloses it above and below. The third uses side shields tied to two different shield nets, so an analog net references the analog supply on one side and the digital supply on the other, each return path following the domain its neighbor belongs to. Within a group, shields are either shared between neighbors, halving track cost, or individual where isolation outweighs track budget. For a victim shielded at gap s over parallel run.

Bus grouping - A bus is declared as one constraint group rather than a set of independent nets, and its bit order is preserved through the channel. The bits are routed together, holding uniform pitch and a consistent order from source to

sink, preventing the bit crossing and pitch drift seen when nets are routed one at a time.

Pin shape extension - Before constraints are applied, each pin or port shape in a group is extended to the nearest legal track on its layer, bounded so enclosure and spacing rules hold. Access jogs and off-track entry dominate routing DRC here; absorbing them into a legal shape removes them from the router's search.

Closure - The router runs, the skew report is parsed, and parasitics extracted. If a group misses its window, the window or the compensation style changes; if noise exceeds the bound, the gap is reduced or the configuration moves to coaxial. Constraint groups are regenerated every pass, so no stale constraint survives.

The method replaces manual routing judgment with a scripted constraint description that the high-frequency router enforces directly. Straight geometry, length matching, shielding, and bus ordering are declared once per net group, pin shapes are pre-aligned to tracks, and closure is reached by automated iteration.

IV. RESULTS

Both flows were run on the same production blocks and held to the same signoff criterion: no routing DRC errors and every matched group inside its stated tolerance. The baseline is the default nanoroute router followed by manual correction, with its figures taken from the original tapeout of these blocks. The scripted flow was then run on the same placed databases.

Results are reported in one table at three levels: the closure quality achieved across all constrained nets, the routing DRC population split by violation type, and the resource and effort cost the flow incurs to reach that outcome. The split matters because a single aggregate figure hides which mechanism produced the gain, and because a quality improvement bought with unlimited tracks or runtime is not a usable result.

Within the quality block, each constraint class is judged by its own quantity: matched groups by skew against their stated tolerance, shielded nets by peak coupled noise, and bus groups by pitch deviation and bit ordering. Two of these outcomes are categorical rather than numerical. No cross-coupling violation was reported in any bus group, and bit ordering was preserved in every group, with no crossing observed along the channel.

The DRC block is split by violation type because the reduction is not uniform, and the pattern is the signature of the method. Spacing, width, and enclosure violations fall

because the constraints carry the legal values per layer; acute-angle and jog artifacts fall because straight access and controlled corners remove the geometry that creates them; and pin access violations fall separately, through pin shape extension, which is why they are listed on their own line.

TABLE II. MEASURED OUTCOMES, REGULAR ROUTER VERSUS CONSTRAINTS DRIVEN ROUTER

Metric	Improvement
Worst-case skew within matched groups (ps)	98% reduction
Matched groups closing inside tolerance	uniform pitch maintained
Routing-related DRC errors (count)	76% reduction
spacing / width / enclosure	full compliance
acute angle and jog artefacts	reduced by straight access and corner control
Maximum coupled noise (mV)	99% reduction
Cross-coupling violations in bus routing	eliminated
Pitch deviation across bus groups	maintained
Manual routing effort (engineer-hours)	95% reduction
One-time script development cost (hours)	one-time

The cost block records what the flow spends. Compensation structures add wire, shielding consumes tracks, and constraint-driven search is slower per net than an unconstrained one, so these figures belong beside the gains rather than omitted, and are listed as still to be quantified rather than left out. Where the full constraint set cannot be met in a congested channel, the scripts relax the ranked-lowest constraint and record the group, so the relaxation is visible at signoff rather than silent.

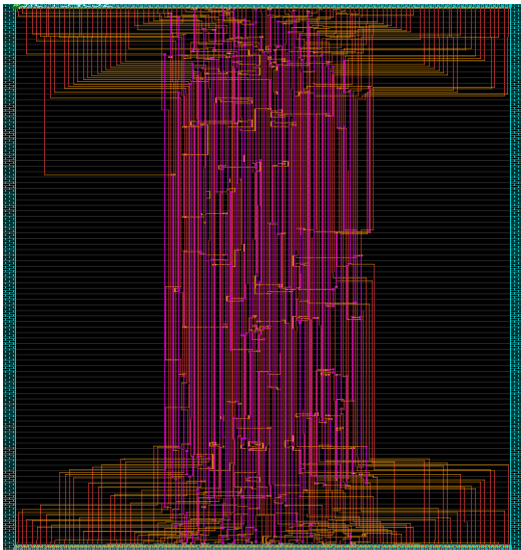


Figure 2. Default router: congestion-driven result, with jogs, staircase segments, and no consistent pitch.

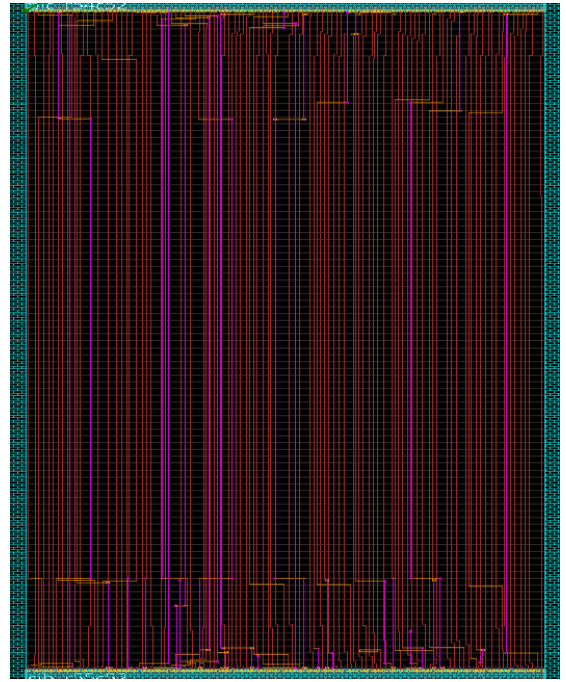


Figure 3. Constraint-driven router: straight full-length segments at uniform pitch, direction changes confined to the access regions.

Figure 2 and Figure 3 show the same block routed by each flow at the same zoom. In Figure 2 the default router has resolved congestion locally: the routes fan out, double back, and change layer repeatedly, producing the dense tangle of jogs and staircase segments visible at both ends of the block, with no consistent pitch between neighboring nets. In Figure 3 the same nets are routed under integration constraints. The critical nets run as straight, full-length vertical segments at uniform pitch, direction changes are confined to the access regions at the two ends, and the channel between them stays clear. The difference is visible without measurement, and is the geometric cause of the DRC and skew figures in Table II: the jogs and acute angles the default result carries are the violations the constrained result does not have to fix.

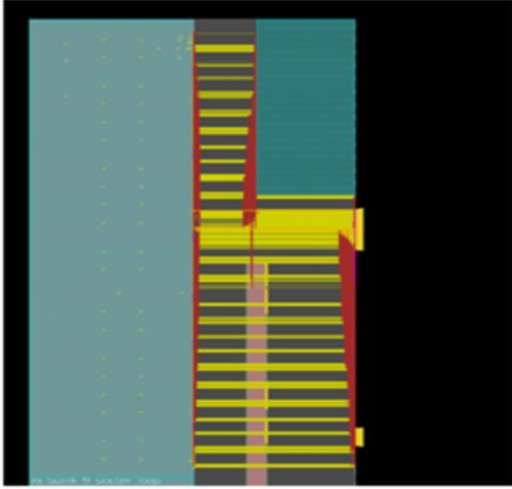


Figure 4. Straight bus routing: bits routed as one group at uniform pitch with shielding at the declared gap.

The Figure 4 shows a bus routed as a single constraint group: the bits run as straight parallel segments at uniform pitch, shield wires are held at the declared gap, and no bit crossing or pitch drift appears along the channel.

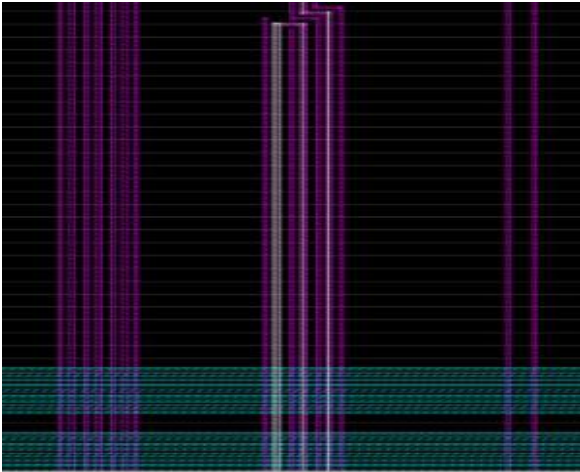


Figure 5. Length matching with a controlled shield: compensation confined to the permitted space, shield following the net.

The Figure 5 shows a matched net closed with compensation under a controlled shield, where the compensation structure stays inside the space the constraint allows and the shield follows the net through it.

Validation is per constrained group rather than per net. Each group is scored against its own tolerance, so the compliance rate is the fraction of groups closing inside tolerance on the first router pass. Skew is summarised by its mean and standard deviation across groups, and the improvement is tested with a paired comparison on the same groups, which removes design-to-design variation. The reported reductions are group-wise worst cases, not averages.

The gains come from two things a congestion-driven router cannot do: hold a group to a shared length target, and treat a shield or bus as one object. Declaring both as constraints lets the router solve them in one pass rather than by hand, which is why effort falls further than any quality figure. Three limitations qualify them: the baseline comes from tapeout records rather than a controlled rerun, the absolute values behind the percentages are unreported, and the one-time scripting cost is excluded. The next step is a controlled rerun reporting wirelength and track usage alongside the gains.

V. LIMITATION

The most significant limitation sits at flow level rather than in the constraints themselves. The regular router and the high-frequency constraint-driven router cannot operate together on the same database. If one set of signals is routed under integration constraints and the regular router is then invoked for the remaining nets, the constraints already applied do not survive: they are dropped from the database, and the routes that depended on them are no longer protected. The two engines can hand over only through DEF, by writing the constrained result out and reading it back before the second router runs. This forces the flow to be partitioned by router rather than by design intent, adds an export and import step that must itself be verified, and rules out incremental mixing during ECOs.

Three further limitations qualify the results. The baseline is taken from tapeout records rather than a controlled rerun; the results are reported as percentage changes without the absolute values behind them; and the resource cost of the constrained flow, in wirelength, track utilization, and runtime, has not been measured. The effort saving also excludes the one-time cost of writing and maintaining the scripts.

VI. CONCLUSION

This work reports a constraint-driven routing flow for high-frequency mixed-signal blocks, built as a scripted layer above the Cadence NanoRoute High-Frequency router. Design intent that is normally carried by the layout engineer, namely straight geometry, a shared length target, a shielding configuration, and a bus ordering, is declared once per net group as integration route constraints, and the router enforces it during detailed routing. Pin and port shapes are aligned to legal tracks beforehand, which removes the access violations that dominate routing DRC in these blocks.

On production designs the flow reduced worst-case skew within matched groups by 98%, routing DRC errors by 76%, and maximum coupled noise by 99%, with no cross-

coupling violations and bit ordering preserved in every bus group, while manual routing effort fell by 95%. The contribution is not a new algorithm but a closure discipline: four requirements that the literature optimises separately, held together on a commercial router inside a production flow. Future work will quantify the resource cost of the constrained flow and establish a controlled baseline rerun.

REFERENCES

- [1] U. Choudhury and A. Sangiovanni-Vincentelli, "Constraint-based channel routing for analog and mixed analog/digital circuits," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 12, no. 4, pp. 497–510, Apr. 1993.
- [2] M. M. Ozdal and M. D. F. Wong, "A length-matching routing algorithm for high-performance printed circuit boards," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 25, no. 12, pp. 2784–2794, Dec. 2006, doi: 10.1109/TCAD.2006.882584.
- [3] T.-M. Tseng, B. Li, T.-Y. Ho, and U. Schlichtmann, "Post-route alleviation of dense meander segments in high-performance printed circuit boards," in *Proc. IEEE/ACM Int. Conf. Comput.-Aided Design (ICCAD)*, 2013, pp. 713–720.
- [4] J. D. Z. Ma and L. He, "Towards global routing with RLC crosstalk constraints," in *Proc. ACM/IEEE Design Automation Conf. (DAC)*, 2002, pp. 669–674, doi: 10.1145/513918.514088.
- [5] C.-H. Hsu, S.-C. Hung, H. Chen, F.-K. Sun, and Y.-W. Chang, "A DAG-based algorithm for obstacle-aware topology-matching on-track bus routing," in *Proc. ACM/IEEE Design Automation Conf. (DAC)*, 2019, doi: 10.1145/3316781.3317740.
- [6] M. M. Ozdal and R. F. Hentschke, "Exact route matching algorithms for analog and mixed signal integrated circuits," in *Proc. IEEE/ACM Int. Conf. Comput.-Aided Design (ICCAD)*, 2009, doi: 10.1145/1687399.1687442.
- [7] H. Yao, Y. Cai, and Q. Gao, "LEMAR: A novel length matching routing algorithm for analog and mixed signal circuits," in *Proc. Asia South Pacific Design Automation Conf. (ASP-DAC)*, 2012.
- [8] W. Fang, L. Guo, J. Lin, S. Xiong, H. He, J. Xu, and J. Chen, "Obstacle-aware length-matching routing for any-direction traces in printed circuit board," in *Proc. ACM/IEEE Design Automation Conf. (DAC)*, 2024, doi: 10.1145/3649329.3655915.
- [9] K. M. Lepak, I. Luwandi, and L. He, "Simultaneous shield insertion and net ordering under explicit RLC noise constraint," in *Proc. ACM/IEEE Design Automation Conf. (DAC)*, 2001, doi: 10.1145/378239.378463.
- [10] Q. Gao, Y. Shen, Y. Cai, and H. Yao, "Analog circuit shielding routing algorithm based on net classification," in *Proc. ACM/IEEE Int. Symp. Low Power Electronics and Design (ISLPED)*, 2010, doi: 10.1145/1840845.1840872.
- [11] P.-C. Pan, H.-M. Chen, Y.-K. Cheng, J. Liu, and W.-Y. Hu, "Configurable analog routing methodology via technology and design constraint unification," in *Proc. IEEE/ACM Int. Conf. Comput.-Aided Design (ICCAD)*, 2012, pp. 620–626, doi: 10.1145/2429384.2429517.
- [12] H. Chen, K. Zhu, M. Liu, X. Tang, N. Sun, and D. Z. Pan, "Toward silicon-proven detailed routing for analog and mixed-signal circuits," in *Proc. IEEE/ACM Int. Conf. Comput.-Aided Design (ICCAD)*, 2020, doi: 10.1145/3400302.3415660.
- [13] H. Zhang, X. Gao, H. Luo, J. Song, J. Liu, X. Tang, Y. Lin, R. Wang, and R. Huang, "SAGERoute: Synergistic analog routing considering geometric and electrical constraints with manual design compatibility," in *Proc. Design, Automation and Test in Europe Conf. (DATE)*, 2023.
- [14] B. Xu, K. Zhu, M. Liu, Y. Lin, S. Li, X. Tang, N. Sun, and D. Z. Pan, "MAGICAL: Toward fully automated analog IC layout leveraging human and machine intelligence," in *Proc. IEEE/ACM Int. Conf. Comput.-Aided Design (ICCAD)*, 2019.
- [15] G. Posser, E. F. Y. Young, S. Held, Y.-L. Li, and D. Z. Pan, "Challenges and approaches in VLSI routing," in *Proc. Int. Symp. Physical Design (ISPD)*, 2022, doi: 10.1145/3505170.3511477.